Document Information

Analyzed document	Computer Fundamentals.pdf (D166063527)
Submitted	2023-05-06 06:50:00
Submitted by	Mumtaz B
Submitter email	mumtaz@code.dbuniversity.ac.in
Similarity	7%
Analysis address	mumtaz.dbuni@analysis.urkund.com

Sources included in the report

W	URL: https://pdfcookie.com/documents/36746371-foundations-of-computing-pradeep-k-sinha-bpb-publicat... Fetched: 2023-01-07 02:19:46	⊞	146
SA	**Operating System-edit.pdf** Document Operating System-edit.pdf (D143421170)	⊞	3
SA	**OS_Book.pdf** Document OS_Book.pdf (D137278070)	⊞	6
SA	**BALU-CSE-Operating System Course Material.pdf** Document BALU-CSE-Operating System Course Material.pdf (D115578367)	⊞	3

Entire Document

Assam Don Bosco University Computer Fundamentals and Windows Based Application

Computer Fundamentals and Windows Based Application

Table of Contents Unit :01 Introduction to Computer System 1.0

Objective 1.1 Introduction 1.2 Classification and Applications 1.3 Overview of the Computer System 1.4 Input and Output Devices 1.5 Personal Computer Configuration 1.6 Computer Languages 1.7 System Softwares 1.8 Summary 1.9 Unit End Questions 1.10 Glossary

Unit :2. Storage Devices 2.0 Objectives 2.1 Introduction 2.2 Memory System in a Computer 2.3 Storage Media 2.4 Random versus sequential access 2.5 Floppy disk, 2.6 Hard disk storage 2.7 Disk Formatting 2.8 Tracks and sectors 2.9 Cylinder 2.10 Summary 2.11 Unit End Questions

Unit :3. Hard disk Drives Interface 3.0 Objectives 3.1 Introduction 3.2 Hard disk Drive interfaces

3.3 RAID 3.4 Optical disk 3.5 CD-ROM and DVD Technology 3.6 Magnetic tape 3.7 Reels, streamers, DAT, DLT, Stripe, 3.8 Smart card 3.9 Modem 3.10 Summary 3.11 Unit End Questions

Unit :4. Input Devices 4.0 Objectives 4.1 Introduction 4.2 Keyboard 4.3 Mouse 4.4 Track Ball 4.5 Track Pad 4.6 Barcode Reader 4.7 Scanner 4.8 OMR 4.9 OCR 4.10 Voice Input 4.11 Video Input 4.12 Digital Camera 4.13 Summary 4.14 Unit End Questions

Unit :5. Output Devices 5.0 Objectives 5.1 Introduction 5.2 Monitors 5.3 Printers 5.4 Plotters

5.5 Summary 5.6 Unit End Questions

Unit :6. Representation of

Number System 6.0

Objectives 6.1 Introduction 6.2

13.17 Printers 13.18 System Tools (Disk Drive Utilities) 13.19 Disk Defragmenter 13.20 Check for Disk Errors (SCAN DISK) 13.21 Backup Your Data 13.22 Clipboard Viewer 13.23 System Information 13.24 NTFS 13.25 Network Neighbourhood (My Network Places) 13.26 Sharing Information between Programs

Unit :14.

Linux Operating System 14.0 Objectives 14.1 Introduction 14.2 Features of Linux Operating System 14.3 Advantages of Linux 14.4 Linux Structure 14.5 Various Flavours of Linux 14.6 Logging in and Logging out from a Linux Session 14.7 Starting a Linux Session: Logging in 14.8 Ending a Linux Session: Logging out 14.9 Types of Users in Linux 14.10 Security for Linux Users: Password 14.11 File System of Linux 14.12 File Naming Conventions in Linux 14.13 Relative Path Name 14.14 A Sample Directory Structure 14.15 Types of Files in Linux 14.16 Linux Commands 14.17 Vi Editor 14.18 The Insert Command and Replacement Command

UNIT 1:

Introduction To Computer Systems 1.0 Objective 1.1 Introduction 1.2 Classification and Applications 1.3 Overview of the Computer System 1.4 Input and Output Devices 1.5 Personal Computer Configuration 1.6 Computer Languages 1.7 System Softwares 1.0 Objectives After going through this unit, we will be in a position to ➢ Define a computer system ➢ Classify computer and Application ➢ Understand the basic organization of computers ➢ Understand the Arithmetic Logical Unit, Control Unit and Central Processing Unit ➢ Define computer memory ➢ Differentiate between input devices and output devices ➢ Distinguish between compiler and interpreter ➢ Describe the use of Interpreter, Assembler, Linker and loader ➢ Differentiate between different types of language 1.1 Introduction Let us begin with the word 'compute'. It means 'to calculate'. We all are familiar with calculations in our day-to-day life. We apply mathematical operations like addition, subtraction, multiplication, etc. and many other formulae for calculations. Simpler calculations take less time. But complex calculations take much longer time. Another factor is accuracy in calculations. So man explored with the idea to develop a machine, which can perform this type of arithmetic calculation faster, and with full accuracy. This gave birth to a device or machine called 'computer'. The computer we see today is quite different from the one made in the beginning. The number of

applications of a computer has increased, the speed and accuracy of calculation has increased. You must appreciate the impact of computers in our day-to-day life. Reservation of tickets in Air Lines and Railways, payment of telephone and electricity bills, deposits and withdrawals of money from banks, business data processing, medical diagnosis, weather forecasting, etc. are some of the areas where computer has become extremely useful. However, there is one limitation of the computer. Human beings do calculations on their own. But computer is a dumb machine and it has to be given proper instructions to carry out its calculation. This is why we should know how a computer works. Definition of Computer Computer is an electronic device, which can do arithmetic calculations faster. For a common man computer is simply a calculator, which works automatic and quite fast. For a person who knows much about it, computer is a machine capable of solving problems and manipulating data. It accepts data, processes the data by doing some mathematical and logical operations and gives us the desired output. Therefore, we may define computer as a device that transforms data. Data can be anything like marks obtained by you in various subjects. It can also be name, age, sex, weight, height, etc. of all the students in your class or income, savings, investments, etc., of a country. Computer can also be defined in terms of its functions. It can accept, store and process data in desired manner, retrieve the stored data as and when required and print the result in desired format. You will know more about these functions as you go through the later lessons. The figure shown below depicts a personal computer. Personal Computer Characteristics Of Computer Let us identify the major characteristics of computer. These can be discussed under the headings of speed, accuracy, diligence, versatility and memory. Speed As you know computer can work very fast. It takes only few seconds for calculations that we take hours

to complete. Suppose you are asked to calculate the average monthly income of one thousand employees in your organization. For this purpose, you have to add income of all employees on a day-to-day basis and find out the average for each one of them. How long will it take for you to do this? One day, two days or one week? Do you know your small computer can finish this work in few seconds? The weather forecasting that you see every day on TV is the results of compilation and analysis of huge amount of data on temperature, humidity, pressure, etc. of various places on computers. It takes few minutes for the computer to process this huge amount of data and give the result. You will be surprised to know that computer can perform millions (1,000,000) of instructions and even more per second. Therefore, we determine the speed of computer in terms of microsecond (10^{-6} part of a second) or nano-second (10^{-9} part of a second). From this you can imagine how fast your computer performs work. Accuracy Suppose some one calculates faster but commits a lot of errors in computing. Such result is useless. There is another aspect. Suppose you want to divide 15 by 7. You may work out up to 2 decimal places and say the dividend is 2.14. I may calculate up to 4 decimal places and say that the result is 2.1428. Some one else may go up to 9 decimal places and say the result is 2.142857143. Hence, in addition to speed, the computer should have accuracy or correctness in computing. The degree of accuracy of computer is very high

accuracy level is determined on the basis of design of computer. The errors in computer are due to human and inaccurate data.
Diligence A computer is
free from tiredness, lack of concentration, fatigue, etc. It can work for hours without creating any error.
If millions of

Due to this capability, it overpowers human being in routine type of work. Versatility It means the capacity to perform completely different type of work. You may use your computer to prepare payroll slips. Next moment you may use it for inventory management or to prepare electric bills.
Power of Remembering Computer has the power of storing any amount of information or data. Any information can be stored and recalled as long as you require it, for any numbers of years. It depends entirely upon you how much data you want to store in a computer and when to lose or retrieve these data. No IQ Computer is a dumb machine and it cannot do any work without instruction from the user. It performs the instructions at tremendous speed and with accuracy. It is you to decide what you want to do and in what sequence. So a computer cannot take its own decision as you can. No Feeling It does not have feelings or emotion, taste, knowledge and experience. Thus it does not get tired even after long hours of work. It does not distinguish between users. Storage The Computer has an in-built memory where it can store a large amount of data. You can also store data in secondary storage devices such as floppies, which can be kept outside your computer and can be carried to other computers. 1.2 Classification and Applications Computers can be classified as per the type of data processed by them or as per size, cost and configuration. As the size and cost of computers is being reduced continuously and configuration is improving at very fast pace, later classification has become important. 1.2.1 Classification of Computers based on Data type This classification of Computers is based on the type of data they are designed to process. Data may be obtained either as a result of counting or through the use of same measuring instrument. Data that are obtained by counting are called discrete data. Examples of discrete data are total number of students in a classroom. This type of data is processed by digital computers. Data that must be obtained through measurement are called continuous data for example speed of an automobile measure by speedometer or the temperature of a patient as measured by a thermometer. Continuous data can be easily processed by analog computer. Digital Computer A digital computer is a counting device that operates on discrete data. It operates by directly counting members (or digits) that represent numerals, letters or other special symbols. Just as digital watches directly count off the seconds and minutes in an hour, digital processors also count discrete values to achieve the desired output results. In contrast to digital processors, however, there are also analog machines that do not compute directly with numbers. They deal with variables that are measured along a continuous scale and are recorded to some predetermined degree of accuracy. Temperature for example may be measured to the nearest tenth of a degree on the Celsius scale, voltage may be measured to the nearest of hundredth of a volt. A service station petrol pump may contain analog processor. Analog Computer Analog computers may be accurate to within 0.1 percent of the correct value or upto limited number of decimal places (e.g. 2-3). But digital computers can obtain degree of accuracy upto required number simply by calculating additional places to the right of the decimal point (e.g. 10-20). For example, a pi(p) has a value of 3.1416. Actually the true value of pi is 3.14159. (this number could go on for pages). For more accuracy larger number of digits are required after decimal place. Desirable features of analog and digital machines are sometimes combined to create a hybrid computing system in a hospital intensive-care unit, for example analog devices may measure a patient's heart function, temperature and other vital signs. These measurements may then be convicted into members and supplied to a digital component in the system. Hybrid Computers Various specifically designed computers are with both digital and analog characteristics combining the advantages of analog and digital computers when working as a system. Hybrid computers are being used extensively in process control system where it is necessary to have a close representation with the physical world. The hybrid system provides the good precision that can be attained with digital computers and the greater control that is possible with analog computers, plus the ability to accept the input data in either form. 1.2.1 Classification of computers based on their size, cost & configuration In this category computers are classified as microcomputers, mini

computers, mainframe computers,

home computers and super computers. Micro Computers: A. Microcomputer is

the smallest general-purpose processing system that can execute program instructions to perform a wide variety of tasks. These systems are typically used by home and school users. They can be divided into different types: 1. Desktop computers - These would typically be supplied with the computer itself (complete with hard disk drive and floppy disk drive) and peripherals such as a screen, a mouse, a keyboard and a CD or DVD drive. a. Advantages: Relatively cheap and easy to add expansion cards to. b. Disadvantages: Can take up a lot of space and are not easily moved. 2. Notebooks (laptops) - These are small (typically 30cm x 20cm), light and easy to carry. The screen is on the inside top flap which hinges open to show the keyboard and mouse controls. They are designed to run on rechargeable batteries or the mains and can contain many of the features available on a desktop computer. Touch pads or a button are usually used to control the screen pointer. a. Advantages: Portable due to their size and ability to run on batteries. b. Disadvantages: Expensive for their processing power compared to desktop computers. You cannot use standard expansion cards. Keyboards and screens not as

good for extended usage. B. Palm-tops & PDAs (personal digital assistants) - These are small hand-held computers. They are usually supplied with software such as a diary, a contacts database, and some form of word processor. Many now have email facilities and even spreadsheets and databases. They either use a small keyboard or touch-sensitive screen and handwriting

recognition software. They can be linked to larger computers directly by cable or through a docking station or using an infra-red link.

a) Advantages: Very portable. b)

Disadvantages: Relatively expensive, limited expansion, non-keyboard versions can be slow to input data. C. Embedded computers - A vast number of modern devices contain some form of built-in computer. Examples include: washing machines, camera, hi-fi systems, telephones, microwave ovens, missiles etc. The inputs are usually sensors so a keyboard may not be needed. The outputs are usually simple displays or motors and relays to control something. The important features of microcomputers are: 1. Microcomputers are known as home computers, personal computers, desktop computers and small business computers. 2. These are small,

low-cost digital computer with a microprocessor as its CPU, a memory unit an input device and an output device. 3. The word length of a microcomputer lies in the range of 8-32 bits. 4.

CPU's are in a single chip

5. Storage capacity is small 6. Applications include general-purpose calculations, industrial control, office automation etc. Personal Computers (PC's):

Personal computers are microcomputers for general-purpose computations. These are classified as PC, PC/XT, PC/AT and super AT (or super micro)

based on their cost and configuration. All these types

contain a CPU, RAM, ROM, CRT display, Keyboard and secondary memory. Features: 1. PC is the simplest and cheapest type of computer

and it uses an INTEL 8088 (microprocessor chip) as CPU, 640 KB, RAM, 8KB ROM (ROM can be extended up to 64 KB) and floppy disk as secondary memory; processing speed of a PC lies in the range 1-5 MIPS (micro-instructions per second). 2. PC/XT (Personal computer extended technology) contains hard disk, floppy disk drives (360/1.2 MB), Intel 8088 CPU, 640 KB RAM and 8 KB ROM. This is costlier than a PC because of the hard disk it contains. PC/XT is a single user system. Processing speed of a

PC lies in the range 1-5 MIPS and hard disk capacity lies in the range 20-80 MB. 3.

PC/AT (Personal computer advanced technology) contains Intel 80286 as CPU, 640 KB RAM (expandable to 4 MB), 64 KB ROM (expandable to 128 KB), hard disk drive and floppy disk drive (360 KB/1.2 MB), its CPU is powerful than the CPU of PC/XT

and four to five terminals can be connected to it to make it multi-user. Processing speed of PC/AT lies in the range 1-5 MIPS and

hard disk capacity lies in the range 20-80 MB. 4. Super AT (or super micros) uses a 32-bit CPU, Intel 80386 & 80486. The RAM capacity of super macros lies in the range 2-8 MB in 80386 CPU can be extended up to 32 MB and in case of 80486 upto 64

MB. The hard disk capacity of super macro lies in the range 200 −800 MB.

The processor speed of 80386 CPU lies in the range 3-5 MIPS and that of 80486 CPU is 12 MIPS

to 25 MHz clock UNIX is used as operating system. The processing speed of home computers and microcomputers for industrial control is less than 1 MIPS. Home computers are used for entertainment, basic education and home management and income tax calculation, investment analysis etc. Mini Computers: The salient features are: -

1. The word length is 32 bits and their

processing speed lies in the range 10-30 MIPS. 2. RAM capacity lies in the range 8MB−96MB (in same it can be extended upto 128-256 MB). 3. The hard disk capacity lies in the range 380 MB – 2GB (Giga Bytes). 4.

Most mini-computers use Motorola 68030 and 68040 CPU. The Minicomputers, which are faster and more powerful than microcomputers, can support upped 64 or even 100 terminals and some minicomputers are uniprocessor and some others are multi-processor systems. These are used for payroll preparation and scientific computation, multi-user and interactive applications in college universities research organizations, industries, sophisticated real time (industrial) control, interactive engineering design work etc. Examples of mini computers are: 1. IBMAS/400/B60 2. VAX8842 (VAX stands for Virtual address - manufactured

by Digital Data Corporation) 3. WIPRO S – 68030V & S-6833V built around 68030 CPU 4.

WIPRO LANDMARK 860 (a super mini) built around Intel i860-CPU 5. HP 9000 series 800 (super mini) built around Hewlett Packard's VLSI RISC processors 6. HCL magnum built around 68030 CPU and 68040 CPU. Mainframe Computers The mainframe computers are faster and more powerful than mini computers. These computers

are used where large amount of data are to be processed or very complex calculations are to be made and these tasks are beyond the computing capacity of

mini computers. The main features are: 1. The word length may be 48, 60 or 64 bits. 2. Memory capacity range is 64-256 MB and hard disk capacity is 1000 MB −10GB or more. 3. Processing speed lies in the range 30-100 MIPS. The mainframe computers

are used in research organizations, large industries, business organizations, government organizations, banks and airline reservations

etc. Example: IBM 4300 series 1. IBM 308X series

2. Latest model IBM 3090 series 3. IBM 9000 series 4. HP 9000 series 8705/400 5. HP 9000 model 8705/300 Supercomputers: The salient features are: 1. Word length is 64-96 bits; memory capacity is 256 MB & more; hard disk capacity 1 GB and more 2. Processing speed lies in the range of 400 MIPS – 10000 MIPS. In a single machine cycle two 64 – bit data can be added

Super computers are much faster and more powerful than mainframe computers.

Super computers are specially designed to maximize the number of FLOPS (Floating Point Operations Per Second). Their FLOPS rating is usually in the range of giga flops. A super computer

contains a number of CPU's, which operate in parallel and

make it faster. They are

used for massive data processing & solving very sophisticated problems like weather forecasting weapons research and development, rocket launching, seismology, atomic, nuclear physics etc. Example of super

computer 1. CRAY – X – MP/14, X-MP/24 & X-MP/48 2. CRAY – MP, CRAYZ, CRAY3, CRAY Y-MPC 3. ETA 10 (Developed by control data Corporation) 4. SX – 2 (Developed by Nippon Electric Corporation, Japan) 5. SX – 3R (25.6 GIGA FLOPS) 6. HITACS – 300 (32 GIGA FLOPS, HITACHI MAKE) 1.3 Basic Computer Operations (Block Structure of Computer) A computer as shown in Figure performs basically five major operations or functions irrespective of their size and make. These are: 1) It accepts data or instructions by way of input 2) It stores data 3) It can process data as required by the user 4) It gives results in the form of output

5) It controls all operations inside a computer. We discuss below each of these operations. 1. Input: This is the process of entering data and programs in to the computer system. You should know that computer is an electronic machine like any other machine, which takes as inputs raw data and performs some processing giving out processed data. Therefore, the input unit takes data from us to the computer in an organized manner for processing. Basic computer Operations 2. Storage: The process of saving data and instructions permanently is known as storage. Data has to be fed into the system before the actual processing starts. It is because the processing speed of Central Processing Unit (CPU) is so fast that the data has to be provided to CPU with the same speed. Therefore the data is first stored in the storage unit for faster access and processing. This storage unit or the primary storage of the computer system is designed to do the above functionality. It provides space for storing data and instructions. The storage unit performs the following major functions: ➢ All data and instructions are stored here before and after processing. ➢ Intermediate results of processing are also stored here. Another storage unit that provides large storage space but slower access time is called secondary storage. Complete memory organization is described in next unit. 3. Output: This is the process of producing results from the data for getting useful information.

| 89% | **MATCHING BLOCK 10/158** | W |

Similarly the output produced by the computer after processing must also be kept somewhere inside the computer before being

given to you in human readable form. Again the output is also stored inside the computer for further processing. 4. Processing: The task of performing operations like arithmetic and logical operations is called processing. The Central Processing Unit (CPU) takes data and instructions from the storage unit and makes all sorts of calculations based on the instructions given and the type of data provided. It is then sent back to the storage unit.

Central Processing Unit (CPU): Central Processing Unit (CPU) controls and performs all operations of the computer system. You may call CPU as the brain of any computer system. It is just like brain that takes all major decisions, makes all sorts of calculations and directs different parts of the computer functions by activating and controlling the operations. This is done by two different units of CPU namely, Arithmetic Logic Unit (ALU) and Control Unit (CU). HARDWARE SOFTWARE Arithmetic Logical Unit (ALU) After you enter data through the input device it is stored in the primary storage unit. Arithmetic Logical Unit performs the actual processing of the data and instruction. The major operations performed by the ALU are addition, subtraction, multiplication, division, logic and comparison. Data is transferred to ALU from storage unit when required. After processing the output is returned back to storage unit for further processing or getting stored. Control Unit (CU) The next component of computer is the Control Unit, which acts like the supervisor seeing that things are done in proper fashion. The control unit determines the sequence in which computer programs and instructions are executed. Things like processing of programs stored in the main memory, interpretation of the instructions and issuing of signals for other units of the computer to execute them. It also acts as a switchboard operator when several users access the computer simultaneously. Thereby it coordinates the activities of computer's peripheral equipment as they perform the input and output. Therefore it is the manager of all operations mentioned in the previous section. 1.4 Input Output Devices A computer is only useful when it is able to communicate with the external environment. When you work with the computer you feed your data and instructions through some devices to the computer. These devices are called Input devices. Similarly computer gives output after processing through other devices called output devices. For a particular application one form of device is more desirable compared to others. We will discuss various types of I/O devices that are used for different types of applications.

100%	**MATCHING BLOCK 2/158**	W

They are also known as peripheral devices because they surround the CPU and

make a communication between computer and the outer world. Input Devices Input devices are necessary to convert our information or data in to a form, which can be understood by the computer. A good input device should provide timely, accurate and useful data to the main memory of the computer for processing. Followings are the most useful input devices. Keyboard: - This is the standard input device attached to all computers. The layout of keyboard is just like the traditional typewriter of the type QWERTY. It also contains some extra command keys and function keys. It contains a total of 101 to 104 keys. A typical keyboard used in a computer is shown in Figure below. You have to press correct combination of keys to input data. The computer can recognize the electrical signals corresponding to the correct key combination and processing is done accordingly. Keyboard Mouse: - Mouse is an input device shown in Figure below that is used with your personal computer.

66%	**MATCHING BLOCK 4/158**	W

It rolls on a small ball and has two or three buttons on the top. When you roll the mouse across a flat surface

the screen censors the mouse in the direction of mouse movement. The cursor moves very fast with mouse giving you more freedom to work in any direction. It is easier and faster to move through a mouse. Mouse Scanner: The keyboard can input only text through keys provided in it. If we want to input a picture the keyboard cannot do that. Scanner is an optical device that can input any graphical matter and display it back. The common optical scanner devices are Magnetic Ink Character Recognition (MICR), Optical Mark Reader (OMR) and Optical Character Reader (OCR).

Magnetic Ink Character Recognition (MICR): - This is widely used by banks to process large volumes of cheques and drafts. Cheques are put inside the MICR. As they enter the reading unit the cheques pass through the magnetic field, which causes the read head to recognize the character of the cheques. Optical Mark Reader (OMR): This technique is used when students have appeared in objective type tests and they had to mark their answer by darkening a square or circular space by pencil.

These answer sheets are directly fed to a computer for grading

where OMR is used. Optical Character Recognition (OCR): - This technique unites the direct reading of any printed character. Suppose you have a set of hand written characters on a piece of paper. You put it inside the scanner of the computer. This pattern is compared with a site of patterns stored inside the computer. Whichever pattern is matched is called a character read. Patterns that cannot be identified are rejected. OCRs are expensive though better the MICR. Output Devices 1. Visual Display Unit: The most popular input/output device is the Visual Display Unit (VDU). It is also called the monitor. A Keyboard is used to input data and Monitor is used to display the input data and to receive massages from the computer. Amonitor has its own box, which is separated from the main computer system and is connected to the computer by cable. In some systems it is compact with the system unit. It can be color or monochrome. 2. Terminals: It is a very popular interactive input-output unit. It can be divided into two types: hard copy terminals and soft copy terminals. A hard copy terminal provides a printout on paper whereas soft copy terminals provide visual copy on monitor. A terminal when connected to a CPU sends instructions directly to the computer. Terminals are also classified as dumb terminals or intelligent terminals depending upon the work situation. Printer: It is an important output device, which can be used to get a printed copy of the processed text or result on paper. There are different types of printers that are designed for different types of applications. Depending on their speed and

approach of printing, printers are classified as impact and non-impact printers. Impact printers use the

familiar typewriter approach of hammering a typeface against the paper and inked ribbon. Dot-matrix printers are of this type. Non-impact

printers do not hit or impact a ribbon to print. They use

electro-static chemicals and ink-jet technologies. Laser printers and Ink-jet printers are of this type. This type of printers can produce color printing and

elaborate graphics. 1.5 personal computer configuration Configuration of a Personal Computer (PC) is specified by two components namely, hardware specification and software packages installed on the system. Specifications of following hardware components of are described in PC: 1. Central Processing Unit (CPU) 2. Computer Memory (RAM and ROM) 3. Data bus 4. Ports 5. Motherboard 6. Hard disk 7. Output Devices 8. Input Devices All these components are inter-connected for the personal computer to work. Details of hardware specifications are described in forthcoming units. Details of various software packages required in running a computer system are described below: Computer Software As you know computer cannot do anything without instructions from the user. In order to do any specific job you have to give a sequence of instructions to the computer. This set of instructions is called a computer program.

Software refers to the set of computer programs, procedures that describe the programs, how they are to be used.

We can say that it is the collection of programs, which increase the capabilities of the hardware. Software guides the computer at every step where to start and stop during a particular job. The process of software development is called programming. You should keep in mind that software and hardware are complementary to each other. Both have to work together to produce meaningful result. Another important point you should know that producing software is difficult and expensive. Software Types Computer software is normally

classified into two broad categories.
➢ Application Software ➢ System software Application Software: Application Software is a set of programs to

carry out operations for a specific application. For example, payroll is an application software for an organization to produce pay slips as an output. Application software is useful for word processing, billing system, accounting, producing statistical report, analysis of numerous data in research, weather forecasting, etc. In later modules you will learn about MS WORD, Lotus 1-2-3 and dBASE III Plus. All these are application softwares. Another example of application software is programming language. Among the programming languages COBOL (Common Business Oriented Language) is more suitable for business application whereas FORTRAN (Formula Translation) is useful for scientific application. We will discuss about languages in next section. System Software: You know that an instruction is a set of programs that has to be fed to the computer for operation of computer system as a whole. When you switch on the computer the programs written in ROM is executed which activates different units of your computer and makes it ready for you to work on it. This set of program can be called

53%	**MATCHING BLOCK 9/158**	W

system software. Therefore system software may be defined as a set of one or more programs designed to control the operation of computer system. System software is a

general program designed for performing tasks such as controlling all operations required to move data into and out of the computer. It communicates with printers, card reader, disk, tapes etc. monitor the use of various hardware like memory, CPU etc. Also system softwares are essential for the development of applications software.

100%	**MATCHING BLOCK 11/158**	W

System Software allows application packages to be run on the computer with less time and effort.

Remember that it is not possible to run application software without system software. Development of system software is a complex task and it requires extensive knowledge of computer technology. Due to its complexity it is not developed in house. Computer manufactures build and supply this system software with the computer system. DOS, UNIX and WINDOWS are some of the widely used system software. Out of these UNIX is a multi-user operating system whereas DOS and WINDOWS are PC-based operating system. We will discuss in detail about DOS and WINDOWS in the next module. So without system software it is impossible to operate your computer. The following picture is shown in Figure relation between hardware, software and you as a user of computer system. Relation between hardware, software 1.6 Computer Languages Language is a system of communication between two persons. Some of the basic natural languages that we are familiar with are English, Hindi, Oriya etc. These are the languages used to communicate among various categories of persons. But how you will communicate with your computer. Your computer will not understand any of these natural languages for transfer of data and instruction. So there are programming languages specially developed so that you could pass your data and instructions to the computer to do specific job. You must have heard names like FORTRAN, BASIC and COBOL etc. These are programming languages. So instructions or programs are written in a particular language based on the type of job. As an example, for scientific application FORTRAN and C languages are used. On the other hand COBOL is used for business applications. There are two major types of programming languages. These are

Low Level Languages and High Level Languages. Low Level languages are

further divided in to Machine language and Assembly language. Low Level Languages The term low level means closeness to the way in which the machine has been built. Low level languages are machine oriented and require extensive knowledge of computer hardware and its configuration. (a) Machine Language Machine Language is the only language that is directly understood by the computer. It does not needs any translator program. We also call it machine code and it is written as strings of 1's (one) and 0's (zero). When this sequence of codes is fed to the computer, it recognizes the codes and converts it in to electrical signals needed to run it. For example, a program instruction may look like this: 1011000111101

It is not an easy language for you to learn because of its difficult to understand. It is efficient for the computer but very inefficient for programmers. It is considered to the first generation language. It is also difficult to debug the program written in this language. Advantage The only advantage is that program of machine language run very fast because no translation program is required for the CPU. Disadvantages 1. It is very difficult to program in machine language. The programmer has to know details of hardware to write program. 2. The programmer has to remember a lot of codes to write a program, which results in program errors. 3. It is difficult to debug the program. (b) Assembly Language It is the first step to improve the programming structure. You should know that computer can handle numbers and letter. Therefore some combination of letters can be used to substitute for number of machine codes. The set of symbols and letters forms the Assembly Language and a translator program is required to translate the Assembly Language to machine language. This translator program is called 'Assembler'. It is considered to be a second-generation language. Advantages: 1. The symbolic programming of Assembly Language is easier to understand and

96%	MATCHING BLOCK 12/158	W

saves a lot of time and effort of the programmer. 2. It is easier to

correct errors and modify program instructions. 3. Assembly Language has the same efficiency of execution as the machine level

69%	MATCHING BLOCK 13/158	W

language. Because this is one-to-one translator between assembly language program and its corresponding machine language program.

Disadvantages: 1. One of the major disadvantages is that assembly language is machine dependent. A program written for one computer might not run in other computers with different hardware configuration. High Level Languages You know that assembly language and machine level language require deep knowledge of computer hardware where as in higher language you have to know only the instructions in English words and logic of the problem irrespective of the type of computer you are using. High level languages are simple languages that use English and mathematical symbols like +, -, %, / etc. for its program construction. High level language has to be converted to machine language for the computer to understand. Compiler and interpreter are used to translate high level language to machine language that can be understood by the computer. Advantages of High Level Languages High level languages have a major advantage over machine and assembly languages that higher level languages are easy to learn and use. It is because that

93%	MATCHING BLOCK 14/158	W

they are similar to the languages used by us in our day to day life. 1.7.

System Softwares The system softwares required to work on computers are compiler, interpreter, assembler, linker and loader. The use of various system softwares is described below: Compiler It is a program translator that translates the instruction of a high level language to machine language. It is called compiler because it compiles machine language instructions for every program instructions of high level language. Thus compiler is a program translator like assembler but more sophisticated. It scans the entire program first and then translates it into machine code. The programs written by the programmer in high level language is called source program. After this program is converted to machine languages by the compiler it is called object program. Higher Level Language—<(Compile)—<Program—< Machine Language Program Compiler Working

100%	MATCHING BLOCK 15/158	W

A compiler can translate only those source programs, which have been written, in

that

will not compile source code written in COBOL language. Object program generated by compiler is machine dependent. It means programs compiled for one type of machine will not run in another type. Therefore every type of machine

one high level language in different machines.
Interpreter An interpreter is another type of program translator used for translating high level language into machine language. It takes one statement ofhigh level languages, translate it into machine language and immediately execute it. Translation and execution are carried out for each statement. It differs from compiler, which translate the entire source program into machine code and does involve in its execution. The advantage of interpreter compared to compiler is its fast response to changes in source program. It eliminates the need for a separate compilation after changes to each program. Interpreters are easy to write and do not require large memory in computer. The disadvantage of interpreter is that it is time consuming method because each time a statement in a program is executed then it is first translated. Thus compiled machine language program runs much faster than an interpreted program. Assembler A computer can understand the program, which is written in machine language. So the program developed in assembly language is required to be converted into the machine language. For this task Assembler is used. Assembler is a program, which converts an assembly language program in machine language program, which is also called object code. An assembler also has subroutines, which are stored into assembler's library. Loader A loader is also system software. It is used for loading and relocating the programs. The loader performs following tasks: ➢ It assigns required memory space to a program, at the time of loading. ➢ To execute the program properly from its memory space, at its load time. ➢ Transferring the program into the assigned memory space. Linker Linker is also system software. It is used to link different program modules to from a large program. All the high level languages have their own inbuilt library of readymade function. These readymade functions are compiled program developed for various standard tasks such as trigonometric function, like sine and cos etc. and are stored in the library of compiler. During the process of linking these program or the functions are linked with the original program. Thus by combining the object of the program and object code of required functions make.exe file. This file is final machine program that can be executed to find output.
SUMMARY ➢ Computer is a machine capable of solving problems and manipulating data. It accepts data, processes the data by doing some mathematical and logical operations and gives us the desired output. ➢ Major characteristics of computer are speed, accuracy, diligence, versatility and memory etc. ➢ Computers are classified into Digital computer, Analog computer and Hybrid computer by the type of data they are designed to process. ➢ Computers classified into Micro, Mini, Mainframe, and Super computer based on their size, cost & configuration. ➢
Personal computers are microcomputers for general-purpose computations. These are classified as PC, PC/XT, PC/AT and super AT (or super micro)
based on their cost and configuration. ➢ A super computer
contains a number of CPU's, which operate in parallel and
make it faster. ➢ They are
used for massive data processing & solving very sophisticated problems. ➢
The major operations performed by the ALU are addition, subtraction, multiplication, division, logic and comparison. Data is transferred to ALU from storage unit when required. After processing the output is returned to storage unit for further processing or getting stored. ➢ The control unit determines the sequence in which computer programs and instructions are executed. Things like processing of programs stored in the main memory, interpretation of

There are two kinds of computer memory: primary and secondary. ➢ A small memory chip is attached between CPU and Main memory whose access time is very close to the processing speed of CPU. It is

called CACHE memory. ➢ Computer software is normally classified into two broad categories. o Application Software o

System software ➢ An interpreter is another type of program translator used for translating high level language into machine language. ➢ Assembler is a program, which converts an assembly language program in machine language program, which is also called object code. ➢ A loader is also system software. It is used for loading and relocating the programs.

➢ Linker is also system software. It is used to link different program modules to from a large program. Questions Define each of the following terms. 1.

What is meant by digital computer? 2. What is computer? Write its characteristics. 3. What are I/O devices? List some important I/O devices. 4. Draw block diagram of computer and explain its all components. 5.

What is the difference between RAM and ROM? 6. What is the function of

Control Unit? 7. Explain Primary and Secondary memory in computer system. 8. What are programming languages, explain difference between Machine, Assembly and High level language. Give some example of high level language 9. What are compiler, how are they different from interpreters. 10. What are Assemblers? 11. What is software? Explain types of software. Glossary Applications software: Software that allows you to perform a task or solve a specific problem. A backup system: A way of storing data in more than one location. A byte: A unit of storage usually made up of eight bits. It represents one character - a letter, digit, or symbol. Central Processing Unit (CPU): Electronic circuits that interpret and execute instructions and communicates with the input, output, and storage devices. Data: Raw unprocessed facts to be processed by the computer. Documentation: Instructions provided with software that includes steps required for installation and use of the product. Freeware: Software considered to be in the public domain and may be used or altered without fee or restriction. Gigahertz (GHZ): A billion machine cycles per second. Hardcopy: Output produced by a printer.

Hardware: The equipment associated with a computer system; it is responsible for performing four basic functions: input, processing, output, and storage. Input devices: Hardware devices that accept data in a form that the computer can utilize; they send the data or instructions to the processing unit to be processed into useful information. A hard disk: An internal disk, a metal platter coated with magnetic oxide that can be magnetized to represent data. A machine cycle: Four steps performed by the central processing unit in carrying out the instructions of a program. A megaflop: One million floating-point operations per second. Megahertz (MHz): One million machine cycles per second. Memory: Primary storage that works with the CPU to hold instructions and data in order to be processed. An office suite: A package of software that contains pieces of software. It is sold as a single package and is designed to work together. Operating systems software: The set of programs that lie between applications software and the hardware devices; it controls the overall activity of a computer. An output device: A hardware device that displays the processed information to the user. Random access memory (RAM): Memory that

the computer user can access.

Read only memory (ROM): Memory that contains programs and data that are permanently recorded when the computer is manufactured. Secondary or auxiliary storage: A more permanent form of storage that does not depend on a constant flow of electricity. Shareware: A form of free software; however, the author of shareware hopes you will make a voluntary contribution for using the product. Softcopy: Monitor output. Software: A program that consists of instructions used to control hardware and accomplish tasks.

UNIT 2: Storage Devices 2.0 Objectives 2.1 Introduction 2.2 Memory System in a Computer 2.3 Storage Media 2.4 Random versus sequential access 2.5 Floppy disk, 2.6 Hard disk storage 2.7 Disk Formatting 2.8 Tracks and sectors 2.9 Cylinder 2.0 Objectives After going through this unit, we will be in a position to: ➢ Define type of storage media and devices ➢ Describe random and sequential data access ➢ Describe floppy disk, hard disk etc. ➢ Define formatting, tracks, sectors, cylinder ➢ Describe speed and storage capacity of disk ➢ Define Primary and secondary memory, RAM, ROM etc. ➢ Define Cache memory 2.1 Introduction As we know that primary memory space is limited in size. RAM is commonly used primary memory. A computer RAM is never large enough to store all the application and program, files, documents, and other file, which are generated in the computer. Secondary storage is required to overcome this need. Secondary storage is also called auxiliary

storage

or mass storage. On secondary storage the data is stored permanently. Once it is stored, we can reload into the computer memory as and when we need it. There are many types of secondary memory devices use different types of storage media. This unit describes complete memory system, various types of storage media and devices.

2.2 Memory System in a Computer There are two kinds of computer memory namely, primary and secondary. This section describes primary memory. Primary memory is accessible directly by the processing unit. RAM is an example of primary memory. As soon as the computer is switched off the contents of the primary memory is lost. You can store and retrieve data much faster with primary memory compared to secondary memory. Secondary memory such as floppy disks, magnetic disk, etc., is located outside the computer. Primary memory is more expensive than secondary memory. Because of this the size of primary memory is less than that of secondary memory. We will discuss about secondary memory later on. Computer memory is used to store two things: i. Instructions to execute a program and ii. Data. When the computer is doing any job. The data that have to be processed, are stored in the primary memory. This data may come from an input device like keyboard or from a secondary storage device like a floppy disk. As program or the set of instructions is kept in primary memory, the computer is able to follow instantly the set of instructions. For example, when you book ticket from railway reservation counter, the com- puter has to follow the same steps: take the request, check the availability of seats, calculate fare, wait for money to be paid, store the reservation and get the ticket printed out. The program containing these steps is kept in memory of the computer and is followed for each request. But inside the computer, the steps followed are quite different from what we see on the monitor or screen. In computer's memory both programs and data are stored in the binary form. You have already been introduced with decimal number system, that is the numbers 1 to 9 and 0. The binary system has only two values 0 and 1. These are called bits. As human beings we all understand decimal system but the computer can only understand binary system. It is because a large number of integrated circuits inside the computer can be considered as switches, which can be made ON, or OFF. If a switch is ON it is considered 1 and if it is OFF it is 0. A number of switches in different states will give you a message like this: 11010110. So the computer takes input in the form of 0 and 1 and gives output in the form 0 and 1 only. Is it not absurd if the computer gives outputs as 0's & 1's only? But you do not have to worry about. Every number in binary system can be converted to decimal system and vice versa; for example, 1010 meaning decimal 10. Therefore it is the computer that takes information or data in decimal form from you, convert it in to binary form, process it producing output in binary form and again convert the output to decimal form.

The primary memory as you know in the computer is in the form of IC's (Integrated Circuits). These circuits are called Random Access Memory (RAM). Each of RAM's locations stores one byte of information. (One byte is equal to 8 bits). A bit is an

acronym for binary digit, which stands for one binary piece of information. This can be either 0 or 1.

You will cone to know more about RAM later. The Primary or internal storage section

78%	MATCHING BLOCK 19/158	W

is made up of several small storage locations (ICs) called cells. Each of these cells can store a fixed number of bits called word length.

Each cell has a unique number assigned to it called the address of the cell and it is used to identify the cells. The address starts at 0 and goes up to (N-1). You should know that the

100%	MATCHING BLOCK 20/158	W

memory is like a large cabinet containing as many drawers as there are addresses

on

87%	MATCHING BLOCK 21/158	W

memory. Each drawer contains a word and the address is written on outside of the drawer.

Capacity of Primary Memory You know that each cell of memory contains one character or 1 byte of data. So, the capacity is defined in terms of byte or words. Thus 64 kilobyte (KB) memory is capable of storing 64 * 1024 = 65536 bytes. (1 kilobyte is 1024 bytes). Amemory size ranges from few kilobytes in small systems to several thousand kilobytes in large mainframe and super computer. In your personal computer you will find memory capacity in the range of 8 MB to 512 MB (MB = Million bytes = 1024 KB) and even 1GB (GB = Giga bytes = 1024 MB). Primary storage is installed inside the computer cabinet and not visible to user. The following types of primary memory are commonly used in a computer system. 1. Random Access Memory (RAM): Random Access Memory (RAM) is most popular primary storage because it is possible to randomly select and use any location of the memory directly for storage and retrieval of data. It takes same time to address any location of the memory. It is also called read/write memory. The storage of data and instructions inside the primary storage is temporary. It disappears from RAM as soon as the power to the computer is switched off. The memories, which loose their content on failure of power supply, are known as volatile memories. So, now we can say that RAM is volatile memory. 2. Read Only Memory (ROM): There is another memory in computer, which is called Read Only Memory (ROM). Again it is the ICs inside the PC that form the ROM. The storage of program and data in the ROM is permanent. ROM stores some standard processing pro- grams supplied by the manufacturers to operate the personal computer. The contents of ROM are readable only as it can not be changed. Important programs like basic input/output opera- tions and power on self test (POST) are stored in ROM. POST examines and initializes vari- ous equipment attached to the PC on power up. The memories, which do not loose their content on failure of power supply, are known as non-volatile memories. ROM is non-vola- tile memory. 3. PROM There is another type of primary memory in computer, which is called Programmable Read Only Memory (PROM). You know that it is not possible to modify or erase programs stored in ROM, but it is possible for you to store your program in PROM chip. Once the programs are written it cannot be changed and remain intact even if power is switched off. Therefore, programs or instructions written in PROM or ROM cannot be erased or changed. 4. EPROM: This stands for Erasable Programmable Read Only Memory, which over come the problem of PROM & ROM. EPROM chip can be programmed time and again by erasing the information stored earlier in it. Information stored in EPROM exposing the chip for some time ultraviolet light and it erases chip is reprogrammed using a special programming facility. When the EPROM is in use information can only be read. 5. Cache Memory: The speed of CPU is extremely high compared to the access time of main memory. Therefore, the performance of CPU decreases due to the slow speed of main memory. To decrease the mismatch in operating speed, a small memory chip is attached
between CPU and Main memory whose access time is very close to the processing speed of CPU. It is called CACHE memory. CACHE memories are accessed much faster than conventional RAM. It is used to store programs or data currently being executed or temporary data frequently used by the CPU. So, each memory makes main memory to be faster and larger than it really is. It is also very expensive to have bigger size of cache memory and its size is normally kept small. 6. Registers: The CPU processes data and instructions with high speed. There is also move- ment of data between various units of computer. It is necessary to transfer the processed data with high speed. So,

| 100% | **MATCHING BLOCK 22/158** | W |

the computer uses a number of special memory units called registers.

They are not part of the main memory,
but they store data or information

| 95% | **MATCHING BLOCK 23/158** | W |

temporarily and pass it on as directed by the control unit. 2.1

Storage Media Secondary storage can store large amount of data in non-volatile memory. This section describes storage media that store desired data. The storage media is the material, which is used to store the data on storage device. Secondary media can be of four types: (1) Magnetic (2) Optical

(3) Magneto-optical (3) Solid State 2.1.1 Magnetic Storage Media As per principle of magnet, when two magnets held one way attract each other and when one of them is reversed, they keep away each other. This effect is caused by polarity. Opposite polarities attract and same polarities repel. These two magnetic states are used to record data on a disk or tape as below. When disk spin or tape moves, electrical signals in the drive's read/write head change the polarity of tiny magnetic particles on the mediums magnetic surface to record 0 and 1. When we open or retrieve the file the process is reversed. The polarities of the medium is immediately under the read/write head induce an electrical current in the read/write head that is transmitted back to the computer in the form of 0 and 1. 2.1.2 Optical Storage Media In optical storage media, data is burnt and removed by using a focused optical beam. CD-ROM and DVD are the devices using this media. There are two types of spaces created by optical beam, namely, pits and lands. The pits are created when the disk surface is forced into mold. The pits are dark and the places without pits are called lands that remain shiny and smooth. A playback device can read these alternating light and dark spots as 0 and 1, when the disk spin in the drive a narrow laser beam is focused on the disk's surface. The amount of the light that is reflected back is determined by whether the beam is focused on a pit or a land. Pits reflect less light then the lands. A device called a photo detector measures the amount of reflected light, which in turn converted into 0 and 1 by read/ write mechanism of the device. 2.1.3 Magneto-Optical Storage Media

It is very popular storage media. These storage systems based on this medium offer high storage capacity and random access. Data is recorded on the disk by heating the surface of the disk by laser. When the disk heated to a specific point, a separate magnetic read/write head can easily change the magnetic orientation of particles. Magneto-Optical (MO) media are available into two formats rewritable and write once and read many (WORM). 2.1.4 Solid State Storage Media This is latest media for data storage called flash memory or memory card. It is available in the market with so many brand names. This memory uses solid state chips much like those used in the computers internal memory (RAM). The data in these chips stays recorded even when the power is turned off, means this type of memory is nonvolatile. The read/write capacity of flash memory is very fast since it does not have any moveable part in it. Devices like digital cameras, digital camcorders, and cell phones may use Compact Flash, Smart Media, or another flash memory card. 2.2 Random Versus Sequentialaccess Data can be stored on the storage devices randomly or sequentially. This is much like the difference between music recorded on a compact disk and music on the cassette tape. The method used to determines how we access the data. On the disk we can directly go to any track and play the music this method is call random access. In case of cassette tape we have to move backward or forward to listen particular music. This method is call sequential data access method. Random Access Memory (RAM) provides true random access to stored data. Irrespective of storage location, time to access data remains the same. However, devices like disks organize their data in such a way that data access is possible in almost equal time. That is why disks are also called direct access devices. Tapes can store data sequentially. That is why it is not very popular. Organization of data in floppy disk and hard disk is discussed below. Both of these disks store data in track and sectors described separately in this unit.

2.3 Floppy Disk It is based on magnetic storage media and similar to magnetic disk discussed above. They are 5.25 inch or 3.5 inch in diameter. They come in single or double density and recorded on one or both surface of the diskette. The capacity of a 5.25-inch floppy is 1.2 mega bytes whereas for 3.5 inch floppy it is 1.44 mega bytes. It is cheaper than any other storage devices and is portable. The floppy is a low cost device particularly suitable for personal computer system. In its design, the floppy disk is similar to the hard drive it operates on the principles of magnetic recording; it uses magnetic heads for data storage and retrieval from the rotating magnetic media. Figure: Floppy Disk Using Floppy Disks Improper preparation or use of a floppy disk can ruin your floppy drive. A few pointers which are kept in mind when use floppy. ➢ Use standard computer disk labels. Note that some labels wrap across the top. Others fit entirely on the front of the disk. ➢ Write on the label if your disks are not kept strictly at home, every label should include your name and something about what's on the disk. (On 3.5" disks, use a felt-tip marker only. Apen or pencil can damage the disk inside.) ➢ Seal all edges down firmly. Aloose corner might stick to the inside of the floppy drive, creating a major mess in there. ➢ Put the label in the right spot. Don't cover the holes in the corners of 3½" floppies. Don't stick to the metal slide. ➢ Insert the floppy right side up. The label goes on top, the round metal part is on the bottom. The edge with the metal slide goes in first.

2.4 Hard Disks These consist of one or more metal platters, which are sealed inside a case. The metal platter consists of thin circular metal sheets with magnetic material coating. The hard disk is usually installed inside the computer's case. These can be removable and cartridge types also. Technically the hard drive is what controls the motion of the hard disks that contain the data. But most people use "hard disk" and "hard drive" interchangeably. They don't make that mistake for floppy disks and floppy drives. It is clearer with floppies that the drive and the disk are separate things. Hard disks provide larger and faster secondary storage capabilities than diskettes. Usually hard disks are permanently mounted inside the computer and are not removable like diskettes. On minicomputers and mainframes, hard disks are often called fixed disks. They are also called direct-access storage devices (DASD). Most personal computers have two to four disk drives. The input/output device that transfers data to and from the hard disk is the hard disk drive. Hard disks are available in different storage capacity in the market. Currently the storage capacity is increasing very fast while the price of storage capacity is reducing. Figure: Hard Disk 2.5 Disk Formatting All magnetic disks are similarly formatted, or divided into areas, called: ➢ Tracks ➢ Sectors ➢ Cylinders

Figure: Hard disk showing internal mechanisms Figure: Hard disk Tracks, Sectors and Cylinder The formatting process sets up a method of assigning addresses to the different areas. It also sets up an area for keeping the list of addresses. Without formatting there would be no way to know what data went with what. It would be like a library where the pages were not in books but were scattered around on the shelves and tables and floors. You'd have a hard time getting a book together. A formatting method allows you to efficiently use the space while still being able to find things. 2.6 Tracks and Sectors A track is a circular ring on one side of the disk. Each track has a number. The diagram shows 3 tracks.

Figure: Showing Tracks on the disk Sectors A disk sector is a wedge-shape piece of the disk, shown in light part. Each sector is numbered. On a 5¼" disk there are 40 tracks with 9 sectors each. On a 3½" disk there are 80 tracks with 9 sectors each. So a 3½" disk has twice as many named places on it as a 5¼" disk. A track sector is the area of intersection of a track and a sector. Figure: Sector on the disk 2.7 Cylinder A cylinder is a set of matched tracks. On a hard disk, a cylinder is made of all the tracks of the same part from all the metal disks that make up the "hard disk". If you put these all together on top of each other, you'd have something that looks like a tin can with no top or bottom - a cylinder.

Figure: Cylinder The computer keeps track of what it has put where on a disk by remembering the addresses of all the sectors used, which would mean remembering some combination of the cylinder, track, and sector. When we format the disk: ➢ All data on the disk is erased. ➢ Surfaces are checked for physical and magnetic defects. ➢ A root directory is created to list where things are on the disk. Summary ➢ Secondary media and devices are can be of three type: o Magnetic

o

Optical o Solid State ➢ There are two kinds of computer memory: primary and secondary. Primary memory is accessible directly by the processing unit ➢ The memories, which loose their content on failure of power supply, are known as volatile memo- ries ➢ The memories, which do not loose their content on failure of power supply, are known as non-volatile memories. ROM is non-volatile memory ➢ A small memory chip is attached

between CPU and Main memory whose access time is very close to the processing speed of CPU. It is called CACHE memory.

Question ➢ What is Volatile memory? ➢ Explain following: o Magnetic storage, o Optical storage, Magneto-optical disk o Solid state ➢ Define the following term: tracks, sector, and cylinder. ➢ What is the difference between primary and secondary memory? Give Example. ➢ Explain Following: o ROM o PROM o EPROM o EEPOM ➢ What do you mean by Random and Sequential Access? ➢ Write a short note on: o Floppy Disk o Hard Disk o Hard disk formatting o Tracks o Sector Cylinder

UNIT 3: Hard Disk Drives Interfaces 3.0 Objectives 3.1 Introduction 3.2 Hard disk Drive interfaces 3.3 RAID 3.4 Optical disk 3.5 CD-ROM and DVD Technology 3.6 Magnetic tape 3.7 Reels, streamers, DAT, DLT, Stripe, 3.8 Smart card 3.9 Modem 3.0 Objectives This unit will give you exposure on ➢ Hard disk interface standards ➢ RAID technology ➢ Optical Disk storage ➢ CD-ROM and DVD technology ➢ Modem functions ➢ Smart Card technology 3.1 Introduction

93%	MATCHING BLOCK 24/158	W

Secondary storage devices need a controller to act as an intermediary between the device and the rest of the computer system. In some

computer

This unit describes commonly used hard disk drive interface standards and companions of hard disk.

3.2 Hard Disk Drive Interface There are three commonly used standards available for interfacing hard disk and its companion devices, namely IDE, EIDE and SCSI. Each one of them is described below: The IDE stands for the abbreviation of the Integrated Device Electronics. In the hard disk drive there is Serial ATA as well as the Parallel ATA. The IDE is the earlier name that is given for the ATA (parallel). There is also other abbreviation of the term IDE called as the Integrated Development Environment. This IDE - Integrated Development Environment is actually a set of programs that are run interface of a single user. Like for example the programming language includes the text several components like the editor, compiler, debugger, etc. All these components are then clubbed together and they have to perform the active functions. In the discussion of the hard disk drive this is the interface type that is used. The hardware interface that is used to connect to the hard disk drive is the IDE - Integrated Device Electronics. This is most commonly used interface for the hard disk drives. Most of the motherboards have the IDE to connect the hard disk drive. There are usually two IDE sockets in the main board or the motherboard. One socket is used for hard disk and other one can be used for the CD drive or another hard disk drive. The data bus from the hard disk drive is plugged in this socket of IDE in the motherboard. This is considered as the most cheaply available interface. The comparison is made with regard to the SCSI interface or any other hard disk drive interface. In the year of the 1980 the IDE was first used as an interface for the hard disk drive. Usually IDE is used for the ATAPI hard disk drives. The ATAPI means the Advanced Technology Attachment Packet Interface. The IDE is also the most popular interface that is used for the other devices like the CD drives or the DVD drives. The ATAPI was actually designed for the CD ROM drives which in turn was improved for interfacing with DVD drives. The IDE was enhanced and the advancement of this interface was named as the EIDE; that is the Enhanced IDE. The ATA standard was also enhanced to include the new versions like the ATA – 2 or the ATA – 3 etc. The ATA – 2 is known as the Fast ATA as its transfer rate is double. The ATA – 4 is named as the Ultra ATA. The technology is named as the UDMA. The UDMA stands for the abbreviation of the Ultra Direct Memory Addressing. These hard disk drives are very faster as compared to the normal hard disk drives. If the speed is of 33 Mega Bytes per second then the name that is given to the ATA is ATA – 33. The EIDE interfaces have the capability to support the hard disk drives with the technology of S.M.A.R.T - Self Monitoring Analysis and Reporting Technology. The technology of the S.M.A.R.T enables to reports its own degradation and enables the user's

operating system to warn if there is any major potential failure. The single interface can be used to connect the two hard disk drives through a single data bus. The two hard disk drives need to use the jumper settings. SCSI (Small Computer Systems Interface) As the IDE is the interface that is used for the hard disk drives; similarly the other type of the interface that is used is the SCSI. The abbreviation of the SCSI stands as Small Computer System Interface. The SCSI is pronounced as 'Scuzzy'. The interface of the SCSI is used since ages to provide the interface for the hard disk drive. It can be considered as an alternative for the parallelATA. Also it can be again considered as an alternative for the old IDE, which was used in the high-end computers. Usually the normal interfaces have the possibility to connect a maximum of three devices. Generally three hard disk drives, or the CD drives and the DVD drives etc. But the interface of the SCSI can provide the advantage to connect even more than the normal range. In other words it is possible to connect more than three hard disk drives. Another important advantage of using the SCSI interface for the hard disk drive is that it is much faster than the regular IDE interface. It may be possible that the hard disk drive is faster but the interface cannot match the speed often hard disk drive. Hence there is no use of the fast hard disk drive in such a situation where in the interface is slow. The speed of the data transmission will remain same. The advantage relating to the SCSI is that the daisy chain of the various different peripherals can be formed. This can be done by utilizing a one host adapter and by utilizing only one slot in the bus. A problem or a disadvantage with the SCSI is that it does not come along with the built in support in any of the computer systems. Therefore if the user is interested in the usage of the SCSI then the computer system should be converted in the format of the SCSI. For the purpose of these conversions, an adaptor card is used. 3.3 Raid Systems RAID stands for Redundant Array of Independent Disks. RAID is an assembly of disk drives, known as disk array that operates as one storage unit. In general, the drives could be any storage system with random data access, such as magnetic hard drives, optical storage etc. When the speed (data transfer rate) is an issue, the fastest SCSI hard drives are typically used.

RAID Functions Redundant Array of Independent Disks (RAID) technology is useful for making data available immediately and recovery of lost data as per level of RAID. Depending on the level of RAID that you are using, this method of data storage provides the data redundancy needed for a highly secure system, with additional benefit of faster retrieval of data through multiple channel access. If one or a few disk drives fail, they can be normally exchanged without interruption of normal system operation. Thus, disk arrays can ensure that no data is lost if one disk drive in the array fails. The array includes drives, controllers, enclosure, power supplies, fans, cables, etc. and software. The host computer can address each array as one drive. There are several types of configurations available in RAID. These configurations are called RAID levels. RAID levels control the ways of organizing data on the drives and organizing the flow of data to and from the host computer. RAID Levels There were 6 RAID levels defined. The numbers used to describe the RAID levels do not imply improves performance, complexity, or reliability. RAID-0 is Striping RAID. It does not, by itself, contribute to fast data access and provides NO redundancy because if one drive fails, all the data in the array will be lost. Two types of RAID provide fast data access for the drives: Mirroring and Parity. Mirroring appeared earlier (in the UC Berkeley Papers) and was originally designated as RAID Level 1. Its main problem is that it requires 100% redundancy or twice as much capacity as was originally needed. On the other hand, its read performance is improved and a higher percentage of drives in a RAID-1 system may fail simultaneously as compared to a Parity RAID system. RAID Level 2 requires the use of non-standard disk drives and is therefore not commercially viable. Parity RAID was identified in the UC Berkeley Papers as RAID Levels 3, 4, 5 and 6. It RAID significantly reduces redundancy overhead to a range of 10% to 33% (compare to 100% for Mirroring RAID-1). RAID Levels 3-5 provide fast data access in case of one disk failure and RAID-6 tolerates failure of two disks at the same time or if the second disk fails later, during the reconstruction period. Levels 1, 3, and 5 are the most commonly used RAID levels and are discussed in the next table. Let us consider a number of drives, say, N drives, connected to the server with MTBF equal to M, and with mean time to repair R.

3.4

Optical Disk

97%   **MATCHING BLOCK 26/158**   W

An optical disk storage system consists of a rotating disk, which is coated with a thin metal or other material that is highly reflective.
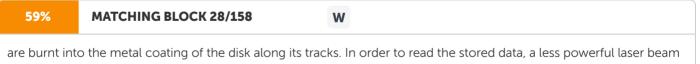

Data recording is done by focusing

a

100%   **MATCHING BLOCK 27/158**   W

laser beam on the surface of the spinning disk. The laser beam is turned on and off at a varying rate because of which tiny

---

holes or pits

59%   **MATCHING BLOCK 28/158**   W

are burnt into the metal coating of the disk along its tracks. In order to read the stored data, a less powerful laser beam focused on the disk surface. This beam is strongly reflected by the coated surface and weakly reflected by the pit, producing patterns on −off reflection that

---

can be converted into electronic signals. The storage density of optical disk is very large and the cost is very low and access time is fast. One small inexpensive disk will be able to store so many magnetic tapes and can be accessed within a few milliseconds. 3.5 CD-ROM and DVD Technology The most common type of optical disk is the CD-ROM. In a CD-ROM, data layer is almost on top of the disc on the label side. While in DVD, the data layer is in the middle of the disc. ADVD can actually have data in two layers. It can access the data from one side or from both sides. This is how a double-sided, double-layered DVD can hold four times data compared to a single-sided and single-layered DVD. Materials The materials used for the data (recording) and metal (reflecting) layers are different for different kinds of optical disks. CD-DVD- Type Data Layer Metal Layer CD-ROM (Audio/video PC software) DVD-ROM (Video/audio, PC use) Read Only Molded Aluminum (Also, silicon, silver, or gold in double-layered DVDs)

CD-R DVD-R DVD+R Recordable (once!) Organic dye Silver, gold, silver alloy CD-RW DVD-RW DVD+RW DVD+RAM Rewritable (Write, erase, write again) Phase-changing metal alloy film Aluminum The most common type of optical disk is the CD-ROM, which stands for Compact Disc - Read Only Memory. It looks just like an audio CD but the recording format is quite different. CD-ROM discs are used for computer software. DVD stands for Digital Video Disk or Digital Versatile Disk. Nowadays, DVD can even store a complete movie. Commercially produced CDs and DVDs are of the Write Once Read Many (WORM) variety. They can't be changed once they are created. The data layer is physically molded into the polycarbonate. Pits (depressions) and lands (surfaces) form the digital data. A metal coating (usually aluminum) reflects the laser light back to the sensor. Oxygen can seep into the disk, especially in high temperatures and high humidity. This corrodes the aluminum, making it too dull to reflect the laser correctly. Hence, CD-ROM and DVD-ROM disks are readable for many, many years if you treat them with respect. The optical disks that you can record on your own computer are CD-R, DVD-R, and DVD+R discs, called writable or recordable disks. In these disks, the metal and data layers are separate. The metal layer can be gold, silver, or a silver alloy. Gold layers are best because gold does not corrode. Naturally, the best is more expensive. Sulfur dioxide can seep in and corrode silver over time. The data layer is an organic dye that the writing laser changes. Once the laser modifies the dye, it cannot be changed again. Write Once! Ultraviolet light and heat can degrade the organic dye. Manufacturers say that these disks have a shelf-life of 5 - 10 years before they are used for recording. There is no testing yet about how long the data will last after you record it. A writable disk is useful as a backup medium when you need long-term storage of your data. It is less efficient for data that changes often since you must make a new recording each time you save your data. Pricing of the disks will be important to your decision to use writable disks. An option for backup storage of changing data is rewritable disks, CD-RW, DVD-RW, DVD+RW. The data layer for these disks uses a phase-changing metal alloy film. This film can be melted by the laser's heat to level out the marks made by the laser and then layered again to record new data. In theory you can erase and write on these disks as many as 1000 times, for CD-RW, and even 100,000 times for the DVD-RW types. Advantages of Optical Disks 1. Physical: An optical disk is much cheaper than tape or a floppy disk. It is physically harder to break or melt or warp 2. Delicacy: It is not sensitive to being touched, though it can get too dirty or scratched to be read. It can be cleaned 3. Magnetic: It is entirely unaffected by magnetic fields. 4. Capacity: Optical disks hold much more data than floppy disks Plus, the non-data side of the disk can have a pretty label! For software providers, an optical disk is a great way to store the software and data that they want to distribute or sell. Care of Optical Disks (CDs, DVDs) Your CDs and DVDs are not going to last forever. They certainly store data longer than floppy disks! Mis-handling your optical disk can quickly make your data unreadable. Even fingerprints can do damage over time. Data loss comes from: ➢ Physical damage - breaking, melting, scratching. ➢ Blocking of laser light by dirt, paint, ink, glue 3.6 Magnetic Tape Magnetic tapes use a positive or negative charge to store information. These charges are read as 0's and 1's. The information on a tape is stored sequentially so this means tape drives are very slow. And most existing tape drives are unable to store as much information as a common hard drive. However, tapes are very inexpensive compared to other types of storage and hold enough files for a good sized back up. ➢ Tapes are Slow and use Sequential Access. ➢ Existing tapes can not compare with the storage capacity of modern Hard Drives. ➢ Are very inexpensive. ➢ Work well for minimal backups. 3.7 Tape Reels and Tape Cartridges There are two types of magnetic tape carriers: tape reels and tape cartridges. Both types can be used for either analog or digital recording. Tape cartridges are normally used only for digital recording.

Tape Reels Tape reels are used on magnetic recorders that use a manually loaded tape supply reel and a separate take-up reel. A reel's purpose is to protect the magnetic tape from damage and contamination. It can be made of plastic, metal, or glass. A reel has two parts, the hub and the flanges. A tape reel is designed to hold magnetic tape on its hub without letting the magnetic tape touch the sides of the flanges. Contrary to popular belief, the flanges are not designed to guide the magnetic tape onto the tape reel. Tape Cartridges Tape cartridges hold a spool of magnetic tape in the same way as tape reels, except that the inside of the cartridge contains both the supply reel and the take-up reel. Unlike tape reels, which must be manually loaded into a recorder, when you insert a tape cartridge into a recorder, it's automatically loaded and ready to use. Figure shows two typical tape cartridges. Figure - Typical tape cartridges DDS/DAT It is abbreviation for Digital Data Storage, the industry standard for Digital Audio Tape (DAT) formats. DAT is a type of magnetic tape that uses a scheme called helical scan to record data. ADAT cartridge is slightly larger than a credit card in width and height and contains a magnetic tape that can hold up to 36 gigabytes (native capacity) of data. Like other types of tapes, DAT is sequential-access media. DLT/SDLT Abbreviation for Digital Linear Tape, DLT is a type of magnetic tape storage device now developed mainly by Quantum Corporation. DLT tapes are ½-inch wide and the cartridges come in several sizes ranging up to 80 gigabytes (native capacity). SDLT (Super Digital Linear Tape) drives available from Dell can hold up to 160 gigabytes (native capacity) of data. LTO Abbreviation for Linear Tape-Open, LTO is the name applied to the Ultrium tape format optimized for high capacity and performance with outstanding reliability. Available in either single or multiple drive configurations, the LTO Ultrium tape format is ideally suited for enterprise-level backup, restore and archive applications. Today, Dell offer LTO drives with capacities ranging up to 400 gigabytes (native

capacity) of data. 3.8 Smart Cards A smart card is a card that is embedded with either a microprocessor and a memory chip or only a memory chip with non-programmable logic. The microprocessor card can add, delete, and otherwise manipulate information on the card, while a memory-chip card (for example, pre-paid phone cards) can only undertake a pre-defined operation. Smart cards, unlike magnetic strip cards, can carry all necessary functions and information on the card. Therefore, they do not require access to remote databases at the time of the transaction. Today, there are three categories of smart cards, all of which are evolving rapidly into new markets and applications: Integrated Circuit (IC) Microprocessor Cards Microprocessor cards (also generally referred as "chip cards" by the industry) offer greater memory storage and security of data than a traditional mag strip card. Chip cards also can process data on the card. The current generation of chip cards has an eight-bit processor, 16KB read-only memory, and 512 bytes of random-access memory. This gives them the equivalent processing power of the original IBM-XT computer, albeit with slightly less memory capacity. These cards are used for a variety of applications, especially those that have built-in cryptography. Cryptography requires manipulation of large numbers. Thus, chip cards have been the main platform for cards that hold a secure digital identity. Some examples of these cards are: Cards that hold money ("stored value cards") Card that hold money equivalents (for example, "affinity cards") Cards that provide secure access to a network Cards that secure cellular phones from fraud Cards that allow set-top boxes on televisions to remain secure from piracy

Integrated Circuit (IC) Memory Cards IC memory cards can hold up to 1-4 KB of data, but have no processor on the card to manipulate data. Thus, they are dependent on the card reader (also known as the card-accepting device) for their processing and are suitable for uses where the card performs a fixed operation. Optical Memory Cards Optical memory cards look like a card with a piece of a CD glued on top - which is basically what they are. Optical memory cards can store up to 4 MB of data. But once written, the data cannot be changed or removed. Thus, this type of card is ideal for record keeping -for example medical files, driving records, or travel histories. 3.9 Modem A modem is a device or program that enables a computer to transmit data over, for example, telephone or cable lines. Computer information is stored digitally, whereas information transmitted over telephone lines is transmitted in the form of analog waves. A modem converts between these two forms. Internal Modem A modem that resides on an expansion board that plugs inside the computer. In contrast, an external modem is a box that attaches to a computer's COM port via cables. External Modem This modem resides in a self-contained box and externally connected to the computer system. Contrast with an internal modem, which resides on a printed circuit board inserted into the computer. External modems tend to be slightly more expensive than internal modems. Many experts consider them superior because they contain lights that indicate how the modem is functioning. In addition, they can easily be moved from one computer to another. However, they do use up one COM port. There is one standard interface for connecting external modems to computers called RS-232. Consequently, any external modem can be attached to any computer that has an RS-232 port, which almost all personal computers have. There are also modems that come as an expansion board that you can

insert into a vacant expansion slot. These are sometimes called onboard or internal modems. The following characteristics distinguish one modem from another: bps How fast the modem can transmit and receive data? At slow rates, modems are measured in terms of baud rates. The slowest rate is 300 baud (about 25 cps). At higher speeds, modems are measured in terms of bits per second (bps). The fastest modems run at 57,600 bps, although they can achieve even higher data transfer rates by compressing the data. Obviously, the faster the transmission rate, the faster you can send and receive data. Note, however, that you cannot receive data any faster than it is being sent. If, for example, the device sending data to your computer is sending it at 2,400 bps, you must receive it at 2,400 bps. It does not always pay, therefore, to have a very fast modem. In addition, some telephone lines are unable to transmit data reliably at very high rates. Voice/data: Many modems support a switch to change between voice and data modes. In data mode, the modem acts like a regular modem. In voice mode, the modem acts like a regular telephone. Modems that support a voice/data switch have a built-in loudspeaker and microphone for voice communication. Auto-answer: An auto-answer modem enables your computer to receive calls in your absence. This is only necessary if you are offering some type of computer service that people can call in to use. Data compression: Some modems perform data compression, which enables them to send data at faster rates. However, the modem at the receiving end must be able to decompress the data using the same compression technique. Flash memory: Some modems come with flash memory rather than conventional ROM, which means that the communications protocols can be easily updated if necessary. Fax capability: Most modern modems are fax modems, which means that they can send and receive faxes. Summary ➢

The hardware interface that is used to connect to the hard disk drive is the IDE - integrated device electronics. ➢ The EIDE interfaces have the capability to support the hard disk drives with the technology of SMART - self monitoring analysis and reporting technology. ➢ The abbreviation of the SCSI stands as small computer system interface. The SCSI is pronounced as 'scuzzy'. The interface of the SCSI is used since ages to provide the interface for the hard disk drive. ➢ RAID is an assembly of disk drives, known as disk array that operates as one storage unit. In general, the drives could be any storage system with random data access, such as magnetic hard drives, optical storage, magnetic tapes, etc. ➢ The most common type of optical disk is the CD-ROM, which stands for compact disc -read only memory. It looks just like an audio CD but the recording format is quite different. CD- ROM discs are used for computer software. ➢ The data layer is physically molded into the polycarbonate. Pits (depressions) and lands (surfaces) form the digital data. ➢ Magnetic tapes use a positive or negative charge to store information. These charges are read as 0's and 1's. The information on a tape is stored sequentially so this means tape drives are very slow. ➢ A smart card is a card that is embedded with either a microprocessor and a memory chip or only a memory chip with non-programmable logic. ➢ A modem is a device or program that enables a computer to transmit data over, for example, telephone or cable lines. QUESTIONS ➢ Explain the Hard disk Interfaces IDE, EIDE and SCSI. ➢ What is RAID? Explain its levels. ➢ Define Pits and lands ➢ What is the difference between CD and DVD? ➢ What is the use of magnetic tape? Explain it type. ➢ What is Smart Card? ➢ What is modem? Explain its types.

UNIT 4: Input Devices 4.0 Objectives 4.1 Introduction 4.2 Keyboard 4.3 Mouse 4.4 Track Ball 4.5 Track Pad 4.6 Barcode Reader 4.7 Scanner 4.8 OMR 4.9 OCR 4.10 Voice Input 4.11 Video Input 4.12 Digital Camera 4.0 Objectives This unit describes ➢ Popular input devices like Keyboard, Mouse etc. ➢ Special purpose input devices like Track ball, Track Pad etc. ➢ Modern input devices like Digital Camera 4.1 Introduction Input devices play key role in communication from human being to computer. By this process, information is taken inward for further processing. A number of input devices have become available for data entry. Depending upon requirement, appropriate device can be interfaced with PC. With introduction of USB port, this interfacing has become very easy. This unit describes popular input devices like keyboard, mouse, modern devices like digital camera and special purpose devices like OMR. 4.2

94%	MATCHING BLOCK 30/158	W

Keyboard Keyboard devices are the most commonly used input devices today. They allow data entry into a computer system by pressing a set of keys (labeled button), which are neatly mounted on a keyboard,

which is connected to the computer system. The most popular keyboard used today is the 101-keys OWERTY keyboard. Fig 4.1 : Keyboard 4.3

94%	MATCHING BLOCK 31/158	W

Mouse As of today, mouse is the most popular point-and-draw device. It has become a must-have input device on personal computer and workstations, which have a GUI-based user interface. Amouse is a small hand- held device, which can comfortably fit in a user's palm. It rolls on a small bearing, and has one or more buttons on the top. When a user rolls the mouse across a flat surface, such as on top the table on which the computer is placed, the graphics cursor moves on the screen of the video display terminal in the direction of the mouse's movement. The graphics cursor, which is used to point and draw, is displayed on the screen as a variety of symbols, such as an arrow, a wrist with pointing finger, etc. Depending on the application, the text and graphics cursor may be displayed on the screen at the same time. Science all movements of the mouse are reproduced by the graphics cursor on the screen, you can move the graphics cursor at a menu item or icon by moving the mouse. For example, move the mouse away from you to move the cursor up on the screen, move the towards you to move the cursor down on the screen, move the mouse right to move the cursor right on the screen, and move the mouse left to move the cursor left on the screen. In this way, the mouse can be used for quick positioning of the graphics cursor on the desired menu item or icon. When positioned at a menu item or an icon, the graphics cursor is said to point to that menu item or icon. With a click of the mouse's button, the system can then be notified of this choice. Note that, notifying the system of a particular choice, out of the various options provided by the software, is much easier with a mouse than pressing various key combinations. With the proper software, a mouse can also be used to draw picture on the screen and edit text. 4.4

Trackball A trackball is a pointing device, which is similar to a mouse. As shown in the ball, which is placed in the base of a mouse, is placed on the top along with the buttons, in case of a trackball. To move the graphics cursor around the screen, the ball is rolled with the fingers. Because the whole device is not be moved for moving the graphics cursor, a trackball requires less space than a mouse for operation. Since it need not be moved for moving the graphics cursor, it is often attached to or built into the keyboard. Track ball built into the keyboard are commonly used in

the

laptop (notebook) computers, because a mouse is not practical for laptop users in a small space. Because the ball is fixed on the top, some people find it helpful to think of a trackball as an

upside-down mouse. Fig. 4.2 : Track Ball

A trackball comes in various shapes and forms with the same functionality. The three commonly used shapes are a ball, and a square. In case of a ball, the ball is rolled with the help of fingers to move the graphics cursor. In case of a button, the button is pushed with a finger in the desired direction of the graphics cursor movement. In case of a square plastic, the fingers is placed on top of it, and moved in the desired direction of the graphics cursor movement. 4.5

Track Pad ALSO CALLED: Touch Pads and Track pads Definition:
A touch pad is a device for pointing (controlling input positioning) on a computer display screen. It is an alternative to the mouse. Originally incorporated in laptop computers, touch pads are also being made for use with desktop computers. Atouch pad works by sensing the user's finger movement and downward pressure.
The
first touch pad was invented by George E. Cepheid in 1988. ... Apple Computer was the first to license and use the touch pad in its PowerBooks laptops in 1994.
The touch pad has since become the leading cursor controlling device in laptops. Many laptops use a trackball. IBM ThinkPad laptops use a "pointing stick" (called a Track Point) that is set into the keyboard. How the Touch Pad Works The touch pad contains several layers of material. The top layer is the pad that you touch. Beneath it are layers (separated by very thin insulation) containing horizontal and vertical rows of electrodes that form a grid. The Track Pad(tm) is a new input device for the Macintosh PowerBook(tm) 500 series computers which replaces the trackball input device used in previous Macintosh PowerBook(tm) computers. The
user study was designed to investigate the impact of the use of the device on the biomechanical load and postural comfort of the users. Since the ease of learning and the achievable performance play an important role for the acceptance of an input device, the study was designed to capture user performance and usage learning progression data. The user study on the Track Pad(tm), a new touch tablet technology input device, was designed to investigate the impact of the use of the device on the biomechanical load and postural comfort of the users. In a one day test, the subjects, experienced Macintosh users, performed tests and worked on tasks, using a portable computer, that were organized to resemble normal office tasks and measure performance. The tasks included intensive use of the keyboard. The performance was measured by text editing tasks and eight Fitts´s Tests with two levels of difficulty. The biomechanical load was measured and evaluated by means of EMG and postural (motion) analyses. General comfort and postural comfort was evaluated with questionnaires.

Touch Screen Touch screen is the most simple, intuitive, and easiest to learn of all input devices. A touch screen enables the users to choose from available options by simply touching with their finger the desired icon or menu item displayed on the computer's screen. Touch screen

are the most preferred human-computer interface devices used in information kiosks. An information kiosk is an unattended system, which is used to information

information as per their requirement. For example, information kiosk may be located. ➢ At an airport or a railway station to provide information to arriving passengers about hotels. Restaurants, tourists spots, etc. in the city. ➢ In large

museums or zoos to guide the visitors to the locations of various attractions and facilities, and to caution them against things,

which they are not supposed to do while inside. 4.6

Bar Code Reader: Data coded in the form of small lines (known as bars) are known as bar codes. Bar codes represent alphanumeric data by a combination of adjacent vertical lines (bars) by varying their width and the spacing between them. They are particularly used for unique identification of all types of goods, books, postal packages, badges, tags, etc. A bar-code reader is a device, which is used for reading (decoding) bar-coded data. It may be a hand- held scanner, or may be embedded in a stationary scanner. It scans a bar-code image, and converts it into an alphanumeric value, which is then fed to the computer to which the bar-code reader is connected, just as though the alphanumeric value had been typed on a keyboard. A bar-code reader uses a laser-beam scanning technology. The laser beam is stroked across the pattern of bars of a bar code. Different bar codes, having different patterns of bars, reflect the beam in different ways, which is sensed by a light-sensitive detector. Reflected light patterns are converted into electrical pulses, and then transmitted to recognition logic circuits, which convert it into an alphanumeric value. Just as there are a variety of internal bit encoding systems, there are a variety of bar-coding systems. The most widely known bar- coding system is the Universal Product Code (UPC), which now appears on almost all retail packages in USA. The UPC, originally used for supermarket items, is now being printed on other consumer goods also. The UPC bar-code patterns are decoded as 10 digits. The first 5 of these digits identify the manufacturer or supplier of the product, and the next 5 digits identify a specific product of the manufacturer. Bar code readers are commonly found in supermarkets and department stores. When a customer picks up a product for purchasing, and brings it to the cash counter for payment, the sales person at the counter uses a bar-code reader to scan the bar code printed on the product. The bar-code

reader converts the bar code into an alphanumeric value, and feeds it to the computer, which looks up the price of the product, possibly updates inventory and sales records, and uses the price and description information to prepare a purchase bill for the customer. Postal delivery services also use bar codes to identify and track each packet. For example Federal Express of USA uses a unique bar-coding system, and their employees can usually tell a customer, within a matter of minutes, the current location of any packet. 4.7

Scanner (Data
Scanning
Devices)

Data scanning devices are input devices, which are used for direct data entry into the computer system from source documents. Some of the data scanning devices are also capable of

the

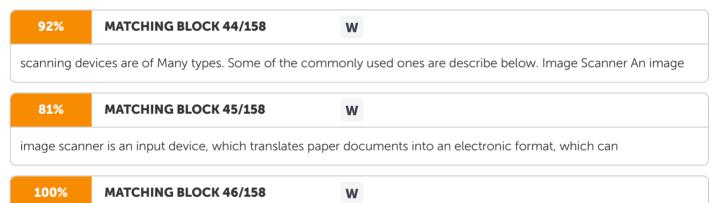recognizing marks or characters. Data scanning devices typically have the following characteristics: ➢ They eliminate some of human effort required to get data into the computer. Human beings do not have to

manually enter

the data. ➢ The reduction in human intervention improves data accuracy and can increase the timeliness of the information processed. ➢ Science scanners are direct data entry devices, they demand high quality of input documents. Documents that are poorly typed or have strikeovers or erasures are normally rejected. ➢ With these devices, form design and ink specification may become more critical than is the case when people key

in the data from the forms. Fig. 4.5 : Flatbed Scanner Fig. 4.6 : Handheld Scanner Data

scanning devices are of Many types. Some of the commonly used ones are describe below. Image Scanner An image

image scanner is an input device, which translates paper documents into an electronic format, which can

can be stored in a computer. The input documents may be typed text, pictures, graphics, or even handwritten material. This input device has been found to be very useful in preserving paper documents in electronic form. The copy of a document stored in a computer in this manner will never deteriorate in quality or become yellow with age, and can be displayed or printed, whenever desired. If the computer in which the scanned document is stored has the right kind of software (called image-processing software), the stored images can be altered and manipulated in interesting ways. Image scanners

come in various shapes and sizes. The two commonly used types

are: 1. Flatbed

scanner. As shown in a flatbed scanner is like a copy machine, which consists of a box having a glass plate on its top and a lid that covers the glass plate. The document to be scanned is placed upside down on the glass plate and moves horizontally from left to right when activated. After scanning one line the light beam moves up a little and scans the next line. The process is repeated for all the lines. It takes about 20 seconds to scan documents of size 21cm *28cm. 2. Hand-held scanner. A hand-held scanner has a set of light emitting diodes encased in a small case, which can be conveniently held in hand during operation. To scan a document, the scanner is dragged from one end of the document to its other end, with its light on. The scanner has to be dragged very steadily and carefully over the document; otherwise, the conversion of the document into its equivalent bit map will not be correct. Due to this reason, hand-held scanners are used only in cases where high accuracy is not needed

in which

Optical Character Recognition (OCR) Device When image scanners are used for inputting text documents (typed or

for

by

which

which

100%   **MATCHING BLOCK 56/158**   W

the computer can interpret as letters, numbers and special characters.

100%   **MATCHING BLOCK 57/158**   W

OCR software is extremely complex, because it is difficult to make a computer recognize an unlimited number

of

91%   **MATCHING BLOCK 58/158**   W

typefaces and fonts. Hence, these software are designed to recognize texts, which are written using standard type fonts (called OCR fonts). Two such standard fonts are OCR-A(American standard) and OCR-B(European standard).

Shows OCR-A fonts. Note that, if the

94%   **MATCHING BLOCK 59/158**   W

document contains italics or bold face letters, or fonts other than that for which the OCR software has been designed, the OCR software will not work effectively. 4.10

Voice Input (Voice Recognition Device) Voice

85%   **MATCHING BLOCK 60/158**   W

recognition devices are input devices, which allow a person to input data to a computer system by speaking to it.

97%   **MATCHING BLOCK 61/158**   W

it. Hence, they make computers much easier to use. However, as a data input device, currently voice recognition systems have limited success, because correct interpretation by a machine of the large number of words in the vocabulary of a language is difficult. The major difficulty has been that people speak with different accents (pronounce differently) and intonations (with different tone or pitch of the voice), and the fact that the meaning of words can vary depending on the context in which they are used. Hence, today's voice recognition systems are limited to accepting few words within a relatively small domain and can be used to enter only limited kinds and quantities of data.

100%   **MATCHING BLOCK 62/158**   W

Although in its infancy, voice recognition systems are already being used for a wide range of applications. Some of its typical applications are as follows: ▪ For inputting data to a computer system by a person in situations where his/her hands are busy, or his/her eyes must be fixed on a measuring instrument or some other object. For example,

doctors

98%   **MATCHING BLOCK 63/158**   W

in an operation room can request certain information about a patient while operating. ▪ For authentication of a user by a computer system based on voice input. ▪ For limited use of computers by individuals with physical disabilities. In addition to making input of data easier, voice recognition systems also provide tremendous freedom of movement to the operator, because the operator is free to stand up and move around, while inputting voice data to the system. 4.11 Video-Input System (

System (Vision-Input System) A vision-input system allows a computer to accept input just by seeing an object. The input data in this case is normally an object's shape and features in the form of an image. The idea is to simulate the capability of a human vision system in a limited sense. A computer with a vision-input device consists of a digital camera, and follows the following steps to recognize a given object: ➢ The digital camera is focused on the input object to take a picture of the object. ➢ The digital camera creates an image of the object in digital form (in 1s and 0s), s.o that it can be stored and interpreted by the computer. ➢ The digitized image of the object is matched against similarly formed pre-stored images in the computer's image database. ➢ Depending on whether a match is found or not, the system takes the appropriate action. Vision-input devices are mainly used today in factories for designing industrial robots,

which

are used for quality" control and assembly processes. For example, a robot used for quality control may inspect objects and separate those, which do not meet certain quality-control specifications. Another robot may be used to separate objects of different shapes in different bins. 4.12

Digital Camera Digital camera is an electronic device used to capture and store photographs electronically in a digital format, instead of using photographic film like conventional cameras, or recording images in an analog format to magnetic tape like many video cameras. Modern compact digital cameras are typically multifunctional, with some devices capable of recording sound and/or video as well as photographs. Summary ➢ Data to be processed by computers originate as written documents. It is necessary to convert these to a form readable by computer's input unit. ➢ Documents are converted to a machine-readable form by entering data using the keyboard of a Personal Computer and storing it in memory. ➢ The commonly used input unit is the keyboard of video terminals. Besides this, magnetic ink character reader, bar code reader and optical mark reader are used for specific applications. Questions ➢ State the necessities of input and output devices.

➢ What are the most extensively used input devices. ➢ Compare the advantages and limitations of a mouse and a key board as data input devices. ➢ What is meant by OMR? Explain its functioning. ➢ Describe an Application for Voice-recognizer. ➢ What are voice-synthesizers? State their importance? ➢ What does MICR stand for? State the functioning of MICR input devices? ➢ What is a mouse? How does it point on the screen? Explain its working? ➢ Is video camera an input device? Justify your answer? ➢ What are touch screen? How do they differ from touch pads? ➢ What are bar codes? Give two applications of bar codes? ➢ Explain how MICR cheques are useful in banks? ➢ What is OCR explain its working?

UNIT 5: Output Devices 5.0 Objectives 5.1 Introduction 5.2 Monitors 5.3 Printers 5.4 Plotters 5.0 Objectives This unit describes ➢ Popular input devices like Keyboard, Mouse etc. ➢ Special purpose input devices like Track ball, Track Pad etc. ➢ Modern input devices like Digital Camera 5.1 Introduction Output devices are the means to obtain results from the computers. This completes the cycle of data processing. Monitors and printers are commonly used output devices. Former provides visual output whereas later provides hard copy of the same. Monitors are available from low resolution to very high-resolution graphics display capability. Printers are also available from dot-matrix good quality laser printers. This unit describes commonly used output devices. 5.2 Monitors The most popular and certainly the most viewed form of output is found on the monitor. Monitor is another term for display screen. The term however usually refers to the entire box whereas display screen can mean just the screen. A monitor is a television-like device used to display data or information. In video display markets, the traditional cathode ray tube (CRT) as used in traditional TVs has logn dominated the market. Traditional CRT technology is analogue technology. It takes generally analogue RGB + sync in, processes it somewhat (possibly some equalizing) and then display it on the screen. Monitors can be classified into following general categories: ➢ Studio video monitors: This kind of monitors have fixed scanning rate for the TV standards in the country in which they are used. This kind of monitors usually feature high quality picture, often high cost, utilitarian case and under scan option. Input is usually composite (i.e., NTSC or PAL) although RGB types are available.

➢ Fixed frequency RGB: This kind of monitors are high resolution fixed rate monitors. Inputs are analog RGB using either separate BNC connectors or a 13W3 (Sun) connector. These often have multiple sync options. This kind of monitors are generally used under scanned for computer workstation (e.g., X-windows) applications so that entire frame buffer is visible. ➢ Multi-scan or auto-scan: This kind of monitors support multiple resolutions and scan rates or multiple ranges of resolutions and scan rates. The quality and cost of these monitors ranges all over the map. Input is most often analog RGB but some older monitors of this type support a variety of digital (TTL) modes as well. A full complement of user controls permits adjustment of brightness, contrast, position, size, etc. to taste. Circuitry in the monitor identifies the video scan rate automatically and sets up the appropriate circuitry. With more sophisticated (and expensive) designs, the monitor automatically sets the appropriate parameters for user preferences from memory as well. The DB15 high density VGA connector is most common though BNCs may be used or may be present as an auxiliary (and better quality) input. ➢ VGA/SVGA/XGA monitors: This kind of monitors is designed to be used with a normal PC graphics cards. This type of monitor is most clearly identified by the fact that is uses DB15 high density VGA connector. The signal input is analog RGB with separate horizontal and vertical sync signals. The basic VGA monitor supported 640x480 resolutions (and some other resolutions near it). Newer SVGA/XGA monitors support a set of higher resolutions (800x600, 1024x768 etc.). Modern PC monitors are generally advanced mortising monitors which can take anything from 640x480 at 60 Hz (around 31 kHz scan rate) up to the highest resolution supported by the monitor (1024x768 up to 1600x1200) at wide variety of refresh rates (refresh rated from 50Hz up to over 100 Hz and horizontal rate from 31 KHz up to well over 60 KHz). There are new digital display technologies to come. Digital display technology is the result of a convergence between digital television and computing. Digital displays accept a digital video data stream and convert it into the technology specific format necessary to drive a display, be it an LCD, PDP, or projector. Those digital display which accept also analogue signals, have A/D converters in their analogue input ports to convert analogue signal to a digital format that rest of the electronics in the display can use. Refresh rate The refresh rate (or "vertical refresh rate", "vertical scan rate" for CRTs) is the number of times in a second that a display is illuminated. This is distinct from the measure of frame rate in that the refresh rate includes

the repeated illumination of identical frames, while frame rate measures how often a display can change from one image to another. For example, a movie projector advances from one frame to the next 24 times each second. But each frame is illuminated twice or three times before the next frame is projected. As a result, the movie projector runs at 24 frames per second, but has a 48 or 72 Hz refresh rate. For Computer Spreadsheets, the term could also be applied to how frequently a cell is updated with a new external value from another source (for example; a Shared public spreadsheet or other external data feed). Here is an overview of different video display resolution standards and de-facto standards in use: Computer Standard Resolution VGA 640 x 480 (4:3) SVGA 800 x 600 (4:3) XGA 1024 x 768 (4:3) WXGA 1280 x 768 (15:9) SXGA 1280 x 1024 (5:4) SXGA+ 1400 x 1050 (4:3) WSXGA 1680 x 1050 (16:10) UXGA 1600 x 1200 (4:3) UXGAW 1900 x 1200 (1.58:1) QXGA 2048 x 1536 (4:3) QVGA (quarter VGA) 320 x 240 (4:3) Analogue TV Standard Resolution PAL 720 x 576 PAL VHS 320 x 576 (approx.) NTSC 640 x 482 NTSC VHS 320 x 482 (approx.) Digital TV Standard Resolution NTSC (preferred format) 648 x 486 D-1 NTSC 720 x 486 D-1 NTSC (square pixels) 720 x 540 PAL 720 x 486 D-1 PAL 720 x 576 D-1 PAL (square pixels) 768 x 576 HDTV 1920 x 1080 Digital Film Standard Resolution Academy standard 2048 x 1536

Color Graphics Adapter (CGA) IBM came back to the fore when color started to appear in computer displays. The CGA standard, introduced in 1981 and primitive by today's standards, was still color; even if only 16 of them. Because the first PCs were for business, the color did not first catch on and the MDA monochrome standard we more often used. As prices came down and clones of the IBM PC were introduced, CGA became more of a standard. The CGA card came with 16 KB of video memory and supported several different modes: ➢ Text mode which included 80x25 text (like the MDA system) in 16 colors. The resolution however was lower as each character was made up of 8x8 pixels instead of the MDA's 9x14 pixels. A 40x25 text mode was also supported in 16 colors. In both, the foreground and background colors could be changed for each character. ➢ Monochrome graphics mode which displayed graphics at 640x200 pixels. This was lower than the Hercules card but seemed to serve the purpose for an initial release and this was quickly replaced with the EGA standard. ➢ Color graphics mode which came in two flavors: a 320x200 pixel mode with four colors and a lesser-used resolution of 160x200 in 16 colors. The four-color mode only had two official palettes to choose from: ➢ Magenta, cyan, white and background color (black by default). ➢ Red, green, brown/yellow and background color (black by default). Video Graphics Array (VGA) With VGA you see a change in the terminology from adapter to array. This was a result of the fact that VGA graphics started to come on the motherboard as a single chip and not as plug-in adapter boards that took up an expansion slot in the computer. While since replaced with other standards for general use, VGA's 640x480 remains a sort of lowest common denominator for all graphics cards. Indeed, even the Windows splash screen logo comes in at 640x480 because it shows before the graphics drivers for higher resolution are loaded into the system. VGA supports both graphics and text modes of operation and can be used to emulate most (but not all) of the EGA, CGA, and MDA modes of operation). The most common VGA graphics modes include: • 640×480 in 16 colors. This is a planar mode with four bit planes. When speaking about VGA, this is the mode most often thought of and is often what is meant when some say "VGA."

• 640×350 in 16 colors. • 320×200 in 16 colors. • 320×200 in 256 colors (Mode 13h). This is a packed-pixel mode. The VGA specification dictated 256KB of video RAM, 16- and 256-color modes, a 262,144 color palette (six bits for each of red, green, and blue), a selectable master clock (25 MHz or 28 MHz), up to 720 horizontal pixels, up to 480 lines, hardware smooth scrolling, split screen support, soft fonts, and more. Another VGA programming trick essentially created another graphics mode: Mode X. By manipulating the 256 KB video RAM four separate planes could be formed where each used 256 colors. Mode X transferred some of the video memory operations to the video hardware instead of keeping them with the CPU. This sped up the display for things like games and was most often seen in 320x240 pixel resolution as that produced square pixels in 4:3 aspect ratio. Mode X also allowed double buffering; a method of keeping multiple video pages in memory in order to quickly flip between them. All VGA 16-color modes supported double buffering; only Mode X could do it in 256 colors. Many other programming tweaks to VGA could (and were) also performed. Some, however, caused monitor display problems such as flickering, roll, and other abnormalities so they were not used commercially. Commercial software typically used "safe" VGA modes. Video memory typically mapped into real mode memory in a PC in the memory spaces. • B0000h (used for monochrome text mode). • B8000h (used for color text and CGA graphics modes). • A0000h (used for EGA/VGA graphics modes). Note that by using the different memory areas it is possible to have two different monitors attached and running in a single computer. Early on, Lotus 1-2-3 took advantage of this by having the ability to display "high resolution" text on an MDA display along with color (low-resolution) graphics showing an associated graph of some part of the spreadsheet. Other such uses included coding on one screen with debugging information on another and similar applications. VGA also had a subset called... Super VGA (SVGA) Super VGA was first defined in 1989 by the Video Electronics Standards Association (VESA); an

association dedicated to providing open standards instead of the closed standards from a single company (IBM). While initially defined as 800x600 with 16 colors, SVGA evolved to 1024x768 with 256 colors and even higher resolutions and colors as time went on. As a result SVGA is more of an umbrella than a fixed standard. Indeed, most any graphics system released between the early 1990s and early 2000s (a decade!) has generally been called SVGA. And, it was up to the user to determine from the specifications if the graphics system supported their needs. The VESA SVGA standard was also called the VESA BIOS Extension (VBE). VBE could be implemented in either hardware or software. Often you would find a version of the VBE in a graphic card's hardware BIOS with extensions in software drivers. How could a standard be so fractured? With the introduction of VGA, the video interface between the adapter and the monitor changed to analog from digital. An analog system can support what is effectively an infinite number of colors. Therefore, color depth largely became a function of how the video adapter was constructed and not the monitor. Therefore, for a set of different monitors there could be thousands of different video adapters that could connect to the monitors and drive them accordingly. Of course, the monitors had to be able to handle the various refresh frequencies and some had to be larger to support the increasing number of pixels but it was easier to produce a few large multi-frequency monitors than it was to produce the graphics computing power necessary to drive them. Thus, while SVGA is an accepted term, it has no specific meaning except to indicate a display capability generally somewhere between 800x600 pixels and 1024x768 pixels at color depths ranging from 256 colors (8-bits) to 65,536 colors (16-bits). But, even those values overlap the various XGA standards... Extended Graphics Array (XGA) IBM's XGA was introduced in 1990 and is generally considered to be a 1024x768 pixel display. It would be wrong, however, to consider XGA a successor to SVGA as the two were initially released about the same time. Indeed, the SVGA "definition" has expanded as seen above and one might consider XGA to have been folded under the SVGA umbrella. Initially, XGA was an enhancement to VGA and added two modes to VGA... • 800x600 pixels at 16-bits/pixel for 65,536 colors. • 1024x768 pixels at 8-bits/pixel for 256 colors. Graphic display processing offloading features from the 8514 system were incorporated into and expanded under XGA. The number and type of drawing primitives were increased over the 8514 and the 16-bit color mode added. Later, and XGA-2 specification added 640x480 at true color, increased the 1024x768 mode to high

color (16-bit/pixel for 65,536 colors) and improved the graphic accelerator performance. Note: XGA was an IBM standard; the VESA released a similar standard called Extended Video Graphics Array (EVGA) in 1991. The two should not be confused. EVGA, as a standalone term, never really caught on. XGA, over time developed into a family of different standards. The following entries summarize this family... Super Extended Graphic Array (SXGA) Super XGA was another step up in resolution and became a family of its own... ➢ Super Extended Graphic Array (SXGA and SXGA+). A resolution of 1280x1024 pixels with an aspect ratio of 5:4 and 1.3 million pixels. This resolution is common in 17-inch to 19- inch LCD monitors. The plus version has a resolution of 1400x1050 pixels with an aspect ratio of 4:3 and 1.47 million pixels. You might find this on notebook LCD screens. Liquid-Crystal Monitors (LCD – monitors) LCD monitors, priorly restricted to notebooks, are now a reality for desktops. It is matter of time until liquid-crystal displays outnumber the traditional monitors in the world. The three greatest advantages of this type of monitor are the use of less space on the desk (especially 17" models or bigger), less power consumption and 100% flicker-free, even with a refresh rate of only 60 frames per second (60 Hz). The main problem with LCD is that they have a native resolution and theoretically the monitor can only work at this resolution. If the LCD is labeled 1024x768, it is very likely that this is its native resolution. If you can't configure your LCD monitor in its native resolution, three things may happen: ➢ The image will be with squared areas, without any definition. ➢ The monitor will centralize the image in the new resolution, reducing the image size and there will a black strip between the image and the screen border. For instance, if the native configuration is 1024x768 and you decreased it to 800x600, this means there are 224 pixels left horizontally and 168 pixels left vertically. The image will be centralized and there will be 112 blanked pixels (black strips) above and below the image and 84 blanked pixels (black strips) to the sides of the image. ➢ The monitor will try to stretch the image so the black area around of the image is not visible, fulfilling the whole screen, using interpolation. However, no interpolation technique is 100% perfect, and you will feel the image has better quality (definition) in the native resolution of the screen. In general, you will feel that the image is slightly out of focus when the monitor is

not in its native resolution. What is going to happen will depend on the monitor model; the choice is the manufacturer's. That's why you have to be careful when buying an LCD monitor, or a notebook one. Preferably, choose one whose native resolution in the one you will feel most comfortable using in comparison to the screen size. There is no point in buying a 14 LCD monitor with 1280 x 1024 native resolution; the image will only be perfectly clear at this resolution, but you won't see almost anything in the screen, because of the reduced size of the icons, menus, etc. The second greatest problem with the liquid-crystal monitors is their performance. The performance is measured in milliseconds, in a feature called response time. The response time measures the monitor speed to draw the screen. If this time is too big (in other words, the monitor is slow), you will see blurry images when you run 3D games in your computer. With LCD monitor with a response time of approximately 25 ms you won't have many problems, but gamemaniacs should look for 16 ms or less. Finally, the third feature of LCD monitors you need to know is called contrast ratio, which measures the brightness difference between the maximum white and the maximum black the monitor can generate. The higher this ration, the better, for you will be able to distinguish between more similar colors. An LCD monitor with 450:1 rate, for instance, is better than a monitor with a 400:1 one. Video Display Controller A Video Display Controller or VDC is an integrated circuit which is the main component in a video signal generator, a device responsible for the production of a TV video signal in a computing or game system. Some VDC's also generate a sound signal, but in that case it's not their main function. VDC's were most often used in the old home-computers of the 80's, but also in some early video game systems. The VDC is always the main component of the video signal generator logic, but sometimes there are also other supporting chips used, such as RAM's to hold the pixel data, ROM's to hold character font's, or perhaps some discrete logic such as shift registers were necessary to build a complete system. In any case, it's the VDC's responsibility to generate the timing of the necessary video signals, such as the horizontal and vertical synchronization signals, and the blanking interval signal. Most often the VDC chip is completely integrated in the logic of the main computer system, (its video RAM appears in the memory map of the main CPU), but sometimes it functions as a coprocessor that can manipulate the video RAM contents independently from the main CPU.

Types of Video Display Controllers Video Display controllers can be (arbitrarily) divided in several different types (here listed from simple to complex); ➢ Video shifters, or "Video shift register based systems" (there is no general agreed upon name for these type of devices) are the most simple type of video controllers, they are, (directly or indirectly) responsible for the video timing signals, but they normally do not access the Video RAM directly. They get the video data from the main CPU, a byte at a time, and convert it to a serial bit stream (hence the technical name "Video shifter"). This serial data stream is then used, together with the synchronization signals, to output a (color) video signal. The main CPU needs to do the bulk of the work. Normally these chips only support a very low resolution Raster graphics mode. ➢ A CRTC, or CRT Controller, generates the video timings and reads video data from a RAM attached to the CRTC, to output it via an external character generator ROM, (for text modes) or directly, (for high resolution graphics modes) to the video output shift register. Because the actual capabilities of the video generator depend for a large degree on the external logic, video generator based on a CRTC chip can have a wide range of capabilities. From very simple (text mode only) systems to very high resolution systems supporting a wide range of colours. Sprites however are normally not supported by these systems. ➢ Video interface controllers are much more complex than CRT controllers, and the external circuitry that is needed with a CRTC is embedded in the video controller chip. Sprites are often supported, as are (RAM based) character generators and video RAM dedicated to color attributes and palette registers (Color lookup tables) for the high-resolution and/or text- modes. ➢ Video coprocessors have their own internal CPU dedicated to reading (and writing) their own video RAM, and converting the contents of this video RAM to a video signal. The main CPU can give commands to the coprocessor, for example to change the video modes or to manipulate the video ram contents. The video coprocessor also controls the (most often RAM based) character generator, the colour attribute RAM, Palette registers and the Spite logic (as long as these exist of course). Video random-access memory (VRAM) Video memory refers generically to various forms of RAM used to store and process image data for a computer monitor or LCD (liquid-crystal display). And even though some people use VRAM (pronounced

"vee-ramm") and video memory interchangeably, VRAM actually refers to a type of chip that can perform reads and writes at the same time, making it faster than DRAM (dynamic RAM). This capability to receive data from the CPU or the graphics accelerator (the video card's built-in processor) at the same time it transmits data to the monitor is why VRAM is often called a dual-ported memory. Because VRAM chips enhance graphics performance, manufacturers often place them on video cards to speed up the creation of on-screen images (but they may also be built into the motherboard on some systems). However, that doesn't necessarily mean that the RAM on your video card is VRAM. Some video cards use some form of conventional DRAM as video memory, such as RDRAM (Rambus DRAM), and others may use WRAM (windows RAM) or SGRAM (synchronous graphics RAM). These video-related RAM technologies are all attempts to improve the performance of the video card. Video cards in use today typically contain between 8MB and 64MB of video memory. To explain how video memory works, let's review the process your computer goes through when you open an application. First, the CPU copies the data from the hard drive and, in the course of processing it, transfers it to DRAM, where it resides while the application remains open. The data is not randomly dumped into DRAM but is situated in a specific location as determined by the OS (operating system). The system bus (network of wires that carries data between the processor and memory) shuttles data between DRAM, the L1 (level 1) and L2 (level 2) caches, and the CPU. While all of this is going on, the CPU continuously transmits display data to the video card. If the video card has a graphics accelerator, the accelerator gathers and processes the data. A small portion of video memory (which could be VRAM if that's what type of memory your video card uses) is allocated to assist the accelerator in this endeavor. The rest of the video memory is used as a frame buffer, where the display data is organized into individual images before appearing on-screen. 5.3

100%	MATCHING BLOCK 66/158	W

Printers Printers are the most commonly used output devices today for producing hard-copy output.

100%	MATCHING BLOCK 67/158	W

The various types of printers in use today are described below. Dot-Matrix Printers Dot-matrix printers are character printers,

which print one character at a time. They

96%	MATCHING BLOCK 68/158	W

form characters and all kinds of images as a pattern of dots. How various types of characters can be formed as a pattern of dots?

Adot-

95%	MATCHING BLOCK 69/158	W

matrix printer has a print head, which can move horizontally (left to right and right to left) across the paper. The print head contains an array of pins, which can be activeted independent of each other to extend and strike against an inked ribbon to form a pattern of dots on the paper. To print a character, the printer activates the appropriate set of pins as the print head, moves horizontally. For faster printing, many dot matrix printers are designed to print both while the print head moves fromleft to right, and while it moves from right to left, on return

shows

Line printer The line printer is a form of high speed impact printer in which one line of type is printed at a time. Print speeds of 600 to 1200 lines-per-minute (approximately 10 to 20 pages per minute) were common. Three principle designs existed: ➢ Drum printers ➢ Chain (train) printers ➢ Bar printers ➢ Comb printers

In a typical drum printer design, a fixed font character set is engraved onto the periphery of a number of print wheels, the number matching the number of columns (letters in a line) the printer could print. The wheels, joined to form a large drum (cylinder), spin at high speed and paper and an inked ribbon are stepped (moved) past the print position. As the desired character for each column passes the print position, a hammer strikes the paper from the rear and presses the paper against the ribbon and the drum, causing the desired character to be recorded on the continuous paper. The character sequences are staggered around the drum, shifting with each column. This avoided the need to fire all of the hammers simultaneously when, for example, a complete line of dashes ("——") needed to be printed. Fig. 9 : Drum Printer Chain printers (also known as train printers) placed the type on moving bars (a horizontally-moving chain). As with the drum printer, as the correct character passed by each column, a hammer was fired from behind the paper. Compared to drum printers, chain printers had the advantage that the type chain could usually be changed by the operator. By selecting chains that had a smaller character set (for example, just numbers and a few punctuation marks), the printer could print much faster than if the chain contained the entire upper- and lower-case alphabet, numbers, and all special symbols. This was because, with many more instances of the numbers appearing in the chain, the time spent waiting for the correct character to "pass by" was greatly reduced. The IBM 1403 is probably the most famous example of a chain printer. Bar printers were similar to chain printers but were slower and less expensive. Rather than a chain moving continuously in one direction, the characters were on fingers mounted on a bar that moved left-to-right and then right-to-left in front of the paper. An example was the IBM 1443. In all three designs, timing of the hammers (the so called "flight time") was critical and was adjustable as part of the servicing of the printer. For drum printers, incorrect timing of the hammer caused the print line to wander up-and-down but the characters were always correctly aligned in their columns. For train and bar printers, incorrect timing of the hammers caused the characters to wander left-and-right of their correct position, but the print line was always level.

Label printer A label printer is a computer peripheral that prints on self-adhesive label material and sometimes cardstock. Label printers are different from ordinary printers because they need to have special feed mechanisms to handle rolled stock, or tear sheet (fanfold) stock. When printing on continuous label stock, there is a tendency for the print location to shift slightly from label to label. To ensure registration of the print area with the target media, label printers use a sensor that detects a gap, notch, line or perforation between labels. This allows the printer to adjust the intake of label stock so that the print aligns correctly with the media. Label printers have a wide variety of applications, including supply chain management, retail price marking, shipping labels, blood and laboratory specimen marking, and fixed assets management. Types of Label Printers ➢ DTP/TTP - There are two types of printing formats. Direct Thermal label Printers use the thermal print head, to print the image directly on to the label. The label is coated with heat sensitive chemicals. Thermal Transfer Printers use an ink ribbon to transfer the image on to the label. These labels usually have a matt finish. TTP's ribbons types have a variety of ink/wax/ resin mixtures. Each ribbon type varies in cost, image durability and color. ➢ Desktop label printers are generally designed for light to medium duty usage with a roll of stock up to 4" wide. They are quiet and inexpensive. ➢ Commercial label printers can typically hold a larger roll of stock (up to 8" wide) and are geared for medium volume printing. ➢ Industrial label printers are designed for heavy duty, continuous operation in warehouses, distribution centers and factories. ➢ RFID readers are specialized label printers that print and encode at the same time on RFID transponders (tags) enclosed in paper or printable synthetic materials. RFID tags need to have printed information for backwards compatibility with barcode systems, and so humans can identify the tag. ➢ Label printer applicators are designed to automate the labeling process. These systems are common in manufacturing and warehousing facilities that require cases and pallets to be labeled for shipping. ➢ Label software is usually PC based and designed to create and/or format labels for printing. The software can use "Windows" drivers or direct drivers embedded in the software. The embedded software drivers are usually faster and better printing control/results.

94%   **MATCHING BLOCK 72/158**   W

Inkjet Printers Inkjet printers are character printers, which form characters and all kinds of images by spraying small drops of ink on to the paper. The print head of an inkjet printer contains up to 64 tiny nozzles, which can be selectively heated up in a few microseconds by an integrated circuit register. When the register heats up, the ink near it vaporizes, and is ejected through the nozzle, and makes a dot on the paper placed in front of the print head. To print a character, the printer selectively heats the appropriate set of nozzles as the print head moves horizontally. Inkjet printers produce higher quality output than dot-matrix printers, because they form characters by very tiny ink dots, A high-resolution inkjet printer has as many as 64 nozzles within height of 7mm, providing print resolution of around 360 dots per inch. Since inkjet printers produce printed output as patterns of tiny dots, they can print any shape of characters, which a programmer can describe. This allows the printer to print many special characters, different sizes of print, and the ability to print graphics, such as charts and graphs.

Fig. 10 : Inkjet Printer

100%   **MATCHING BLOCK 73/158**   W

Inkjet printers are non-impact printers, because they print by spraying ink on the paper. Hence, they are quiet in operation. However, like other non-impact printers, inkjet printers cannot be used to produce multiple copies of a document in a single printing. Inkjet printers

---

100%   **MATCHING BLOCK 74/158**   W

are slower than dot-matrix printers with speeds usually ranging between 40 to 300 characters per second. Typically, an inkjet printer is more expensive than a

---

dot-matrix printer.

100%   **MATCHING BLOCK 75/158**   W

They are preferred by individuals and organizations for generating printed outputs, if the speed of printing is not an important factor.

Laser Printers Laser printers are page printers, which print one page at a time. The main components of a laser printer are a laser beam source, a multi-sided mirror, a photoconductive drum and toner (tiny particles of oppositely charged ink). To print a page of output, the laser beam is focused on the electro statically charged drum by the spinning multi-sided mirror. The mirror focuses the laser beam on the Surface of the drum in a manner to create the patterns of characters/images to be printed on the page. As the drum is photoconductive, a difference in electric charge is created on those parts of the drum surface, which are exposed to the laser beam. Asa result, the toner, which is composed of oppositely charged ink particles, sticks to the drum in the places, where the laser beam has charged-the drum's surface. The toner is then permanently fused on the paper with heat and pressure to generate the printed output. The drum is then rotated and cleaned with a rubber blade to remove the toner sticking to its surface to prepare the drum for the next page printing. Fig. 11 : Laser

Printers

Laser printers produce very high quality output, because they form characters by very tiny ink particles. The most common laser printers have resolution of 600 dpi (

dots per inch), whereas some high-end laser printers have resolution

of 1200 dpi. Because of their high resolution, these printers give excellent graphics art quality. Since laser printers produce printed output as patterns generated by the laser beam, they can print any shape

of

characters, which a programmer can describe. This allows the printer to print many special characters, different sizes of print, and the ability to print graphics, such as charts and graphs. Laser printers are non-impact printers, because they do not have hammers striking on an inked ribbon or paper. Hence, they are very quiet in operation. However, like other

non-impact printers, laser printers cannot be used to produce multiple copies of a document in a single printing. Laser printers

are faster in printing speed than other printers discussed before. Low speed laser printers can

print 4

to 12 pages per minute. Very high-speed laser printers are also available, which can print 500 to 1000 pages

per

minute. That is fast enough to print this entire book in about one minute. Because of their better print quality

and

printing speed, laser printers are more expensive than other printers.

Thermal Wax (Thermal-transfer Printers) An inexpensive alternative to the ink-jet printer is the thermal-transfer printer which uses heat to transfer into to paper. A thermal-transfer printer produces near letter-quality characters. These printers bond the ink onto the paper by heating pins which press against a special ink ribbon. Thermal-transfer printers can produce colour printouts by using a color ribbon. Thermal printers are inexpensive and are used in most calculators and many FAX machines. These types of printers produce low-quality print, and the paper tends to curl and fade after a few weeks or months. 5.4

Plotters We saw above that dot matrix, inkjet and laser printers are capable of producing graphics output. However, many engineering design applications, like architectural plan of a building, design of mechanical components of an aircraft or a car,, etc., often require

high-quality,

perfectly-proportioned graphic output on large sheets. The various types of printers discussed above are not suitable for meeting this output requirement of such applications, A special type of output device, called plotters, is used for this purpose. Plotters are ideal output device for architects, engineers, city planners, and others who need to routinely generate high- precision, hard-copy,

and

graphic out of widely varying sizes. The two commonly used types of plotters are drum plotter and flatbed plotter. They are briefly described below. Drum Plotter In a drum plotter the paper, on which the design has to be made is placed over a drum,

which

can rotate in both clockwise and anti-clockwise directions to produce vertical motion. The mechanism also consists of one or more pen holders mounted perpendicular to the drum's surface. He pen(s) clamped in the holders(s) can move left to right, or right to left

of

produce horizontal motion. The movements of the drum and the pen(s) are controlled by the graph-plotting program. That is, under computer control, the drum and the pen(s) move simultaneously to draw the designs and graphs on the sheet placed on the drum. The plotter can also annotate the designs and graphs so drawn by using the pen to draw characters of various sizes. Since each pen is program selectable, pens having ink of different colors can be mounted in different holders to produce multi- colored designs. Flatbed Plotter A flatbed plotter plots a design or graph on a sheet of paper,

which

that is spread and fixed over a rectangular flatbed table. In this type of plotter, normally the paper dose not move, and the pen holding mechanism is designed to provide all types of motions necessary to draw complex designs and graphs. That is, under computer control, the pen(s) move in the required manner to draw complex designs and graphs on the sheet placed on the flatbed table. The plotter can also annotate the designs and graphs so drawn by using the pen to draw characters of various sizes. Here also, prevision is there to mount more than one pen in the pen(s) holding mechanism. Since each pen is program selectable, pens having ink of different colors can be mounted in different holders to produce multi-colored designs. The plot size is restricted by the area of the bed. Some may be as small as A4 size (8"*11"page), while some very large beds used in the design of cars, ships, aircrafts, buildings, highways, etc. can be up to 20 ft. by 50 ft. some plotters are also designed to etch plastic or metal plates. In this case, the plastic or metal sheet is spread on the bed, and the drawing pen has a sharp-edged needle.

Inkjet Plotter with Encad Quality An inkjet plotter from Encad offers superior line drawing and brilliant color printing performance, in a compact design, with patented snap-on ink caddy reservoirs for quick, clean installation. Transparent caddies keep ink levels always visible in this outstanding inkjet plotter PEN PLOTTER: If you use AutoCAD® or any other CAD Program, chances are you could use FPLOT™ If you are a software developer who wants a simple but powerful method of including vector graphics in your programs, FPLOT is your answer. FPLOT is a pen plotter emulator, which means that you can view or print files created for 'a pen plotter. Most CAD programs produce excellent pen plotter files, so the best way to print your CAD plots is with FPLOT. FPLOT features ➢ 100% implementation of the HP-GL/2 pen plotter language. ➢ Fast high resolution printing of your plot files without having to start your CAD program. ➢ No jagged lines. ➢ Screen preview with zoom and pan. ➢ Full color printing support. ➢ Full control over pen colors and pen widths. ➢ Full control over fonts. ➢ Lets you paste all or part of your plot to Windows programs, including word processors. ➢ Full online help. ➢ Works with AutoCAD® and all other CAD programs. ➢ Runs under Windows 95/98/Me/NT/2000/XP. Electrostatic: Multimedia kit Electrostatic printers use electrical charges transferred to a nonconducting surface that either attract or

repel the toner. There are several types of electrostatic processes: direct electrostatic, color xerography and Electro Ink. Direct Electrostatic

Direct Electrostatic printers apply a charge directly to specially coated print media. Liquid toner particles are then swept across the paper and stick to the charged regions. Repelled toner is removed from the page before the next color pass. After all colors have been placed, the toner is then fused. This technology can be easily modified for large format printing. Liquid toner provides the advantage of finer toner particles that can be used to achieve high resolution output. Color Xerography Color xerography uses a pre-charged drum or belt that conducts a charge only when exposed to light. The scanning laser is used to discharge this belt or drum which creates an invisible image. Toner containing small iron particles are magnetically attracted to the appropriate areas of the image and repelled from others. This image is then transferred to a roller which collects all four colors. The image is then electrostatically transferred to plain paper where it is fused by heat and pressure. The QMS magi color printer line is an example of a color xerography printer. ElectroInk ElectroInk is a variation of xerography that uses liquid toner. The liquid toner is charged electrostatically and brought into contact with the photoconductor where it is either attracted or repelled. The colors are imaged to an offset blanket from which the composite color image is then transferred to the paper media. This liquid toner offers the advantage of delivering very small dots that can produce very high resolutions. The Indigo E-Print 1000 is based upon this technology. Summary ➢ Flat bed optical scanners are used to convert printed material to bit maps. If printed material uses standard fonts a character recognition software is used to convert the bit map files to ASCII files. ➢ Devices with removable and portable storage media such as floppy disks, CD-ROMs and solid state memory cards are also used as I/O units. ➢ The most common output units are line printers and character printers. These provide hard copies. Video display terminals may also be used to examine sample outputs. Inkjet and

laser printers are used to get good quality printed outputs. ➢ Graphical output maybe obtained using plotters, inkjet or laser printers. Video graphical display units are also useful to examine graphical outputs. ➢ Graphical data may also be input using light-pens associated with video terminals or by using flat-bed graphics tablets. QUESTIONS ➢ How does Analog monitor differ from digital monitor. ➢ What are the different types of Printers? ➢ What are the criteria for classification of printer? ➢ What is plotter and how does it work? ➢ What are the thermal printers and hoe does it works? ➢ What is a monitor and how does it work? ➢ Which device can provide us with graphical out put? ➢ What are the major chrematistics of a monitor? ➢ What are the different video standards? ➢ What is a laser printer? Explain its working?

UNIT 6: Representation of
Number System 6.0
Objectives 6.1 Introduction 6.2
Number System 6.2.1 Decimal
Number System 6.2.2 Binary Number System 6.2.3 Octal Number System 6.2.4 Hexadecimal Number System 6.3 Conversion
to decimal number system 6.3.1 Binary
to Decimal 6.3.2 Octal to Decimal 6.3.3 Hexadecimal to Decimal 6.4 Conversion to other number system from decimal system 6.4.1 Decimal to Binary 6.4.2 Decimal to Octal 6.4.3 Decimal to Hexadecimal 6.5 Other Conversions 6.5.1 Binary to Octal and vice-versa 6.5.2 Binary to Hexadecimal
and vice-versa 6.6 Unsigned and Signed Integers 6.7 Floating Point Representation 6.8 Definitions 6.8.1 Bit 6.8.2 Byte 6.8.3 Machine Word 6.9 Boolean Algebra 6.9.1 Fundamental concepts of Boolean algebra 6.9.2 Postulates of Boolean algebra 6.9.3 Minimization of Boolean Functions 6.9.4 Simplification of Boolean equations 6.9.5 Karnaugh map (K-map)
6.0
Objectives
After going through this unit you will be able to: ➢ Learn
how data is represented in the computer ➢ Understand various number system i.e. positional and non positional ➢ Learn to convert number from one number system to another. 6.1 Introduction Reports prepared by any personnel contains symbols from the set of alphanumeric characters like A-Z,a- z,0-9,+,- ..... Computer cannot understand all these characters because computer is an electronic device. It is a bistable device means with two states "On" and "Off". Therefore, computer deals with data converted into simplest form which can be processed electronically, that is in binary form, where it substitutes "On" with "1" and "Off" with 0. To store and process data in a binary form, a coding scheme had been devised to represent characters as standardized methods. Data stored in a coded form is basically of three types – Numeric, Alphabetic and Alphanumeric. Every computer stores these data types in a coded form i.e. in binary other number system. 6.2 Number System Number system is of two types: a)

73%	**MATCHING BLOCK 91/158**	W

Non-Positional Number System In this system there are symbols used as I for 1, II for 2, III for 3

and so on is known as Roman

numbers, therefore, this leads to the positional number system development. b) Positional Number System In this system, digits

i.e. decimal number system contains 10 digits (i.e. 0 to 9) having base 10. Therefore number in this system is repre-sented by (number) base i.e. $254 = (254)_{10}$.

Computing the value of a positional number ➢ A number = Digit x Base (position of the digit) i.e. $(254)_{10} = 2 \times 10^2 + 2 \times 10^1 + 4 \times 10^0$. Hence the value of a number can be viewed as the sum of the positional values of the symbols in the number. The rightmost symbol in the number has the lowest weight, whereas the leftmost symbol in the number has the highest weight. Hence the rightmost and the leftmost symbols in a number are also called the least significant digit (LSD) and the most significant digit (MSD) respectively. The following table shows the base and base set values of different number system: Number System Base Base Set Largest 4 digit number

Decimal 10 0,1,2,3,4,5,6,7,8,9 9999 Binary 2 0,1 1111 Octal 8 0,1,2,3,4,5,6,7 7777 Hexadecimal 16 0,1,2,3,4,5,6,7,8,9, A, B,C,D,E,F

FFFF 6.2.1 Decimal

Number System There are 10 digits available in decimal number system as shown in the above table. It is known as Base 10 system. Therefore the number 798 in this system is represented by $(798)_{10}$. $798 = 7 \times 10^2 + 9 \times 10^1 + 8 \times 10^0$ 6.2.2 Binary Number System It contains two unique digits 0 and 1. It is known as Base 2 system. The number thus formed is the combination of these two digits such as $(1101)_2$. 3- Bit numbers with their decimal values Binary Number Decimal Equivalent 000 0 001 1 010 2 011 3 100 4

101 5 110 6 111 7 Why Binary?

use

under: ➢
Information is handled in a

off (0). All types of information are

represented within the computer by the presence or absence of these types of signals. The binary number system, which has only two digits (0 and 1), is most suitable for expressing the two possible states. ➢

Computer circuits only have to handle two binary digits, this greatly simplifies the internal circuit design of computers. ➢

Everything that can be done in decimal number system can also be done in binary number system.

Binary Arithmetic As in decimal number system, binary arithmetic involves mainly four operations like Addition, Subtraction, Multiplication and Division. The operations of each are described below: A. Addition To perform addition on binary numbers following rules may be applied: 0 + 0 = 0 0 + 1 = 1 1 + 0 = 1 1 + 1 = 10 or 0 with carry over 1 Example Carries 10001 11101 29 +10001 +17 ————— ——- 101110 46 Example

11001 25 + 110 + 6 ————— ——- 11111 31 Example 111001 57 +110111 +55 ————— ——- 1110000 112 B. Subtraction To perform subtraction on binary numbers following rules may be applied: 0 − 0 = 0 1 − 0 = 1 1 − 1 = 0 0 − 1 = 1 with one as borrow Simple subtraction Example 11011 27 - 1001 ————— 10010 - 9 18 ——- Example 11001 25 - 1010 - 10 ————— ——-1111 15 Subtraction of Binary number by 2's compliment method 1's compliment of a binary number is obtained by flipping the each digit i.e. making all 0's to 1's and all 1's to 0's. 2's compliment of a binary number can be obtained by adding 1 to its 1's compliment. While performing subtraction on binary numbers following steps may be followed: ➢ Both the numbers should be of the same size. If not, then make both the numbers of the same size by adding leading zeros. ➢ Compliment the number to be subtracted by flipping the each digit i.e. changing 1's to 0's and 0's to

1's. The compliment of 1001 is 0110. ➢ Perform binary addition as above i.e. Add the first number to the result obtained in the above step. ➢ If the result of the addition is increased by one digit means there is a carry of 1, then remove this (ignore the left most digit) and add to the final result. ➢ If the size of the result is not increased (equal to the original size), then re-compliment the answer and attach a negative (-) sign in front of the number. Example

C. Multiplication The multiplication of binary numbers follows the same convention as of decimal number system. The result may be obtained by sequence of additions and shifts.

Example D. Division The division in binary number system is also same as of in decimal number system. It may be performed as sequence of subtraction and shifts. Example: 11100 divided by 100 (28 ÷4) 111 Quotient 100√11100 100 0110 100 0100 100 000 Remainder Therefore answer is 111 6.2.3 Octal Number System In octal number system, there are 8 unique digits available in octal number system. These are from 0 −7. Thus, any number formed by these digits is combination of these digits. It is known as Base 8 system. The value of a digit in a number depends upon its position in the number. Therefore, the number 402 of octal number system can be expressed as: 6.2.4 Hexadecimal Number System In Hexadecimal Number System, there are 16 digits available. These are from 0 − 9 and A, B, C, D, E, F where A denotes 10, B denotes 11 ... and F denotes 15. Thus, any number formed is combination of these digits. It is known as Base 16 system. The value of a digit in a number depends upon its position in the number. Therefore, the number B58 of hexadecimal number system can be expressed as: (B58) 16 = B (11) x 16 2 + 5 x 16 1 + 8 x 16 0 = 11 x 256 + 80 + 8 = 2816 + 88 = (2904) 10 The hexadecimal notations are used not only to represent numbers, but also used to represent binary numbers in compact form. This is so because in most of the computers data occupy multiple of 4 bits that is equivalent to single hexadecimal digit.

6.3 Conversion to

Decimal

Number System Usually numbers expressed in decimal number system are much more meaningful to us,

because

we have been using decimal numbers in our daily routine life. Any number in one number system can be represented in any other number system.

There are many methods, which can be used to convert numbers from one base to another.

The
following steps are required to convert a number from any base to base 10:-
For conversion of a number from any number system to decimal number system (
base 10), multiply each of digit of the number by (Base Value) position of the digit and then add the result. i.e. ➢

Determine the column(positional) value of each digit. This depends on the position of the digit and the base of the number system. ➢ Multiply the obtained column values by the digits in the corresponding columns. ➢ Sum the products calculated in the above step. The total is the equivalent value in decimal.

Fractional Numbers In any number system, fractional numbers are formed in the same general way as in the decimal number system. For example, in the decimal number system $0.275 = (2 \times 10^{-1}) + (7 \times 10^{-2}) + (5 \times 10^{-3}) = 0.2 + 0.07 + 0.005$ 6.3.1

Binary
to Decimal For conversion of a number from binary number system to decimal number system,
follow the above procedure i.e. multiply each of binary digit by 2 position of the digit and then add the result. Example $(10101)_2 = (?)_{10}$ Step − 1: Determine the
column values Column Number (From Right) Column Value 1 2 0 = 1 2 2 1 = 2 3 2 2 = 4 4 2 3 = 8 5 2 4 = 16

Step - 2: Multiply column values by corresponding column digits 16 8 4 2 1 x 1 x 0 x 1 x 0 x 1 16 0 4 0 1 Step − 3: Sum the products 16+0+4+0+1 = 21 or $1 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0$ = 16 + 0 + 4 + 0 + 1 = 21 Hence $(10101)_2 = (21)_{10}$ Example $(101.101)_2 = (?)_{10}$ $(101.101)_2 = 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 + 1 \times 2^{-1} + 0 \times 2^{-2} + 1 \times 2^{-3}$ = 4 + 0 + 1 + 1/2 + 0/4 + 1/8 = 5 + 0.5 + 0 + 0.125 = $(5.625)_{10}$ 6.3.2

Octal to Decimal For conversion of a number from octal number system to decimal number system,
multiply each of octal digit by 8 position of the digit and then
add the result. Example $(3057.51)_8 = (?)_{10}$ $(3057.51)_8 = 3 \times 8^3 + 0 \times 8^2 + 5 \times 8^1 + 7 \times 8^0 + 5 \times 8^{-1} + 1 \times 8^{-2}$ = 3 x 512 + 0 + 40 + 7 + 5/8 + 1/64 = 1536 + 47 + 0.625 + 0.015625 $(3057.51)_8 = (1583.640625)_{10}$ 6.3.3 Hexadecimal
to Decimal
Similarly, for conversion of a number from hexadecimal number system to decimal number system,
multiply each of hexadecimal digit by (16) position of the digit and then add the result. Example
$(C1A.2C)_{16} = (?)_{10} = C \times 16^2 + 1 \times 16^1 + A \times 16^0 + 2 \times 16^{-1} + C \times 16^{-2}$ = 12 x 16 2 + 16 + 10 x 1 + 2/16 + 12/256 = 3072 + 26+0.125+0.046875 $(C1A.2C)_{16} = (3098.171875)_{10}$ 6.4 Conversion to Other Number System From Decimal System As decimal number may contain only integer or integer part along with fractional part, thus calculation shall be done in two parts.

The following steps are used to convert a number from decimal to another base: Integer Part i- Divide the decimal number by the value of the new base. ii- Record the remainder

iii- Repeat the step – i with the quotient and then step – ii, until the quotient becomes 0 or less than the value of the new base. Fractional Part i- Multiply the fractional part by the value of the new base. ii- Record the integer part, if it exists, else record 0 iii- Repeat the step – i with the result of the previous multiplication and then step – ii, until the fractional part becomes 0. In case of infinite calculations, generally 6 digits are taken. 6.4.1 Decimal to Binary Example (30) 10 = (?) 2 Here the new base is 2. 30/2 = 15 Remainder 0 15/2 = 7 Remainder 1 7/2 = 3 Remainder 1 3/2 = 1 Remainder 1 or 2 30 2 15 0 2 7 1 2 3 1 1 1

Now start writing in the order from the last obtained till the first remainder. Thus the binary equivalent of (30) 10 = (11110) 2 Example (30.625) 10 = (?) 2 Convert integer part i.e. 30 into binary as above = 11110 Convert fraction part 0.625 into binary as below: 0.625 x 2 = 1.25 take away integer part and record = 1 0.25 x 2 = 0.50 take away integer part and record = 0 0.50 x 2 = 1.00 take away integer part and record = 1 Thus binary equivalent of fraction part is 101. Hence (30.625) 10 = (11110.101) 2 6.4.2 Decimal to Octal (952) 10 = (?) 8 Here the new base is 8. 952/8 = 119 Remainder 0 or 119/8 = 14 Remainder 7 14/8 = 1 Remainder 6 Now start writing in the order from the last obtained till the first remainder. Thus the octal equivalent of (952) 10 = (1670) 8 6.4.3 Decimal to Hexadecimal (952) 10 = (?) 16 Here the new base is 16. 952/16 = 59 Remainder 8 or 59/16 = 3 Remainder 11 8 952 8 119 0 8 14 7 8 1 6 16 952 16 59 8 3 11 (B)

Now start writing in the order from the last obtained till the first remainder. Thus the hexadecimal equivalent of (952) 10 = (3B8) 16 6.5

OTHER CONVERSIONS The following steps are used to convert a number from a base other than 10, to a base other than 10

i.e. binary to octal, hexadecimal and vice-versa: i.

Convert the original number to a decimal number (base 10) ii. Convert the decimal number obtained

in step-I to the new base number 6.5.1

Binary to Octal and vice-versa (101010) 2 = (?) 8 Step – i : Convert 101010 to base 10 (101010) 2 = 1 x 2 5 + 0 x 2 4 + 1 x 2 3 + 0 x 2 2 + 1 x 2 1 + 0 x 2 0 = 32 + 0 + 8 + 0 + 2 + 0 = (42) 10 Step – ii: Convert (42) 10 to base 8 42/8 = 5 remainder 2 5/8 = 0 remainder 5 Hence (42) 10 = (52) 8 Therefore (101010) 2 = (42) 10 = (52) 8 Shortcut method for Binary to Octal Conversion A binary number is easily converted to a octal number by dividing the bits of the binary number into groups of 3-bits. This is because of the fact

that the maximum value of one digit is equal to the maximum value of three digits in binary. Therefore, the value of one octal digit is equivalent to 3

bits of binary. Binary Coded Octal Numbers Octal Number Binary Coded Octal Number Decimal Equivalent 0 000 0 1 001 1

2 010 2 3 011 3 4 100 4 5 101 5 6 110 6 7 111 7 (101010) 2 = (?) 8 Divide the bits into group of 3 from right as 101 - 010 As per the above table, 010 is equivalent to 2 in octal number and 101 is equivalent to 5 in octal number. Therefore octal equivalent of the given binary number is (52) 8 Hence (101010) 2 = (52) 8 Which is same as converted above. Now for quick conversion of octal to binary, each digit of octal number be converted into its 3-bits of its binary equivalent. 6.5.1 Binary to Hexadecimal and vice-versa (101010) 2 = (?) 16 Step − i : Convert 101010 to base 10 (101010) 2 = 1 x 2 5 + 0 x 2 4 + 1 x 2 3 + 0 x 2 2 + 1 x 2 1 + 0 x 2 0 = 32 + 0 + 8 + 0 + 2 + 0 = (42) 10 Step − ii: Convert (42) 10 to base 16 42/16 = 2 remainder 10 (A) 2/16 = 0 remainder 2 Hence (42) 10 = (2A) 16 Therefore (101010) 2 = (42) 10 = (2A) 16 Shortcut method for Binary to Octal Conversion A binary number is easily converted to a hexadecimal number by dividing the bits of the binary number into groups of 4-bits. This is because of the fact

bits of binary.

Binary Coded Hexadecimal Numbers Hexadecimal Number Binary Coded Octal Number Decimal Equivalent 0 0000 0 1 0001 1 2 0010 2 3 0011 3 4 0100 4 5 0101 5 6 0110 6 7 0111 7 8 1000 8 9 1001 9 A 1010 10 B 1011 11 C 1100 12 D 1101 13 E 1110 14 F 1111 15 (101010) 2 = (?) 16 Divide the bits into group of 4 from right as 0010 - 1010 As per the above table, 1010 is equivalent to Ain hexadecimal number and 0010 is equivalent to 2 in hexadecimal number. Therefore hexadecimal equivalent of the given binary number is (2A) 16 Hence (101010) 2 = (2A) 16 Which is same as converted above. Now for quick conversion of hexadecimal to binary, each digit of octal number is converted into its 4-bits of its binary equivalent. 6.6 Unsigned and Signed Integers An integer is a number with no fractional part; it can be positive, negative or zero. In ordinary usage, one uses a minus sign to designate a negative integer. However, a computer can only store information in bits, which can only have the values zero or one. We might expect, therefore, that the storage of negative integers in a computer might require some special technique. Consider a single digit decimal number: in a single decimal digit, you can write a number between 0 and 9. In two decimal digits, you can write a number between 0 and 99, and so on. Since nine is equivalent to 10 1 - 1, 99 is equivalent to 10 2 - 1, etc., in n decimal digits, you can write a number between 0 and 10 n - 1.

2 Analogously, in the binary number system, An unsigned integer containing n bits can have a value between 0 and 2 n - 1 (which is 2 n different values). When a computer performs an unsigned integer arithmetic operation, there are three possible prob- lems which can occur: 1. If the result is too large to fit into the number of bits assigned to it, an "overflow" is said to have occurred. For example if the result of an operation using 16 bit integers is larger than 65,535, an overflow results. 2. In the division of two integers, if the result is not itself an integer, a "truncation" is said to have occurred: 10 divided by 3 is truncated to 3, and the extra 1/3 is lost. This is not a problem, of course, if the programmer's intention was to ignore the remainder! 3. Any division by zero is an error, since division by zero is not possible in the context of arithmetic. Signed Integers Signed integers are stored in a computer using 2's complement. As you recall, when computing the 2's complement of a number it was necessary to know how many bits were to be used in the final result; leading zeroes were appended to the most significant digit in order to make the number the appropriate length. Since the process of computing the 2's complement involves first computing the 1's complement, these leading zeros become leading ones, and the left most bit of a negative number is therefore always 1. In computers, the left most bit of a signed integer is called the "sign bit". Consider an 8 bit signed integer: let us begin with 0 0 0 0 0 0 0 0 2 and start counting by repeatedly adding 1: When you get to 127, the integer has a value of 0 1 1 1 1 1 1 1 2 ; this is easy to see because you know now that a 7 bit integer can contain a value between 0 and 2 7 - 1, or 127. What happens when we add 1? ➢ If the integer were unsigned, the next value would be 10000000 2 or 128 (2 7). But since this is a signed integer, 10000000 2 is a negative value: the sign bit is 1! ➢ Since this is the case, we must ask the question: what is the decimal value corresponding to the signed integer 1 0 0 0 0 0 0 0 2 ? To answer this question, we must take the 2's complement of that value, by first taking the 1's complement and then adding one. ➢ The 1's complement is 0 1 1 1 1 1 1 1, or decimal 127. Since we must now add 1 to that, our conclusion is that the signed integer 1 0 0 0 0 0 0 0 2 must be equivalent to decimal -128!

Odd as this may seem, it is in fact the only consistent way to interpret 2's complement signed integers. Let us continue now to "count" by adding 1 to $1\,0\,0\,0\,0\,0\,0\,0_2$: $\succ$ $1\,0\,0\,0\,0\,0\,0\,0_2 + 0\,0\,0\,0\,0\,0\,0\,1_2$ is $1\,0\,0\,0\,0\,0\,0\,1_2$. $\succ$ To find the decimal equivalent of $1\,0\,0\,0\,0\,0\,0\,1_2$, we again take the 2's complement: the 1's complement is $0\,1\,1\,1\,1\,1\,1\,0_2$ and adding 1 we get $0\,1\,1\,1\,1\,1\,1\,1_2$ (127) so $1\,0\,0\,0\,0\,0\,0\,1_2$ is equivalent to -127. $\succ$ We see then that once we have accepted the fact that $1\,0\,0\,0\,0\,0\,0\,0_2$ is decimal -128, counting by adding one works as we would expect. $\succ$ Note that the most negative number which we can store in an 8 bit signed integer is -128, which is -2^{8-1} , and that the largest positive signed integer we can store in an 8 bit signed integer is 127, which is $2^{8-1}-1$. $\succ$ The number of integers between -128 and + 127 (inclusive) is 256, which is 2^8 ; this is the same number of values which an unsigned 8 bit integer can contain (from 0 to 255). $\succ$ Eventually we will count all the way up to $1\,1\,1\,1\,1\,1\,1\,1_2$. The 1's complement of this number is obviously 0, so $1\,1\,1\,1\,1\,1\,1\,1_2$ must be the decimal equivalent of -1. Using our deliberations on 8 bit signed integers as a guide, we come to the following observations about signed integer arithmetic in general: $\succ$ If a signed integer has n bits, it can contain a number between -2^{n-1} and $+(2^{n-1}-1)$. $\succ$ Since both signed and unsigned integers of n bits in length can represent 2^n different values, there is no inherent way to distinguish signed integers from unsigned integers simply by looking at them; the software designer is responsible for using them correctly. $\succ$ No matter what the length, if a signed integer has a binary value of all 1's, it is equal to decimal -1. You should verify that a signed short integer can hold decimal values from -32,768 to +32,767, a signed long integer can contain values from -2,147,483,648 to +2,147,483,647 and a signed double integer can represent decimal values from -9, 223, 372, 036, 854, 775, 808 to +9,223,372,036,854,775,807. 6.7 Floating Point Representation

The floating-point representation of a number needs two parts. The first part represents a signed, fixed-point number called the mantissa. The second part designates the position of the decimal (or binary) point is called the exponent. The fixed point mantissa may be a fraction or an integer. For

example, the decimal number +6132.789 is represented in floating point as follows: The mantissa has a 0 in the leftmost position to denote a plus. The mantissa here is considered to be a fixed-point fraction, so the decimal point is assumed to be at the left of the most significant digit. The decimal mantissa, when stored in a register, requires at least 29 flip-flops. This representation is equivalent to the number expressed as a fraction times 10 to an exponent, that is +.6132780 x 10 +04 , because of this analogy; the mantissa is sometimes called the fraction part. Floating point is always interpreted to represent a number in the following form: $m \times r^e$ Only the mantissa m and the exponent e are physically represented in the register (including their signs). The radix r and the radix-point position of the mantissa are always assumed. Arithmetic operations with floating-point numbers are more complicated than arithmetic operations with fixed-point numbers and their execution takes longer and requires more complex hardware. However, floating point representation is must for scientific computations. 6.8 Definitions 6.8.1 Bit Acronym for Binary digit i.e. 0 and 1. It stands for one binary piece of information. 6.8.2 Byte

A fixed number of adjacent bits that represent a particular character or symbol. Normally a byte consists of eight bits. 6.8.3

Machine Word In computing, a group of bits or characters that occupies one or more storage locations and is treated by computers as a unit. Synonym Computer Word. 6.9 Boolean Algebra Boolean algebra is an

---

81% **MATCHING BLOCK 111/158** W

algebra, which deals with the binary number system. It is very useful in designing logic circuits,

---

which are used by

81% **MATCHING BLOCK 112/158** W

the processors of computer systems. Boolean algebra provides an economical and straightforward approach to the design of relay, and other types of switching circuits.

Boolean algebra is used extensively in designing the circuit

94% **MATCHING BLOCK 113/158** W

Boolean algebra is used extensively in designing the circuitry used in computers. 6.9.1 Fundamental concepts of Boolean Algebra $\succ$

Use of binary digits (0 and 1), variables used in the Boolean equations may assume only two possible value (0 and 1). $\succ$

Logical Addition : - The symbol "+" is used for logical addition operator. It is also known as "OR" operator. We can define the + symbol by listing all possible combinations of A and B, and the resulting value of C, in the equation A + B = C. Since, the variables A and B can have only two possible values (0

and 1), only four (2 2) combinations of input are possible as shown

in the table below. Such a table is known as a truth table.

Inputs Output A + B = C 0 0 0 0 1 1 1 0 1 1 1 1 Truth table for the

logical OR operator ➢ Logical Multiplication: - The symbol "." is used for logical multiplication operator. It is also known as "AND" operator. We can define the "." symbol by listing all possible combinations of A and B, and the resulting value of C, in the equation A.B = C.

Inputs Output A . B = C 0 0 0 0 1 0 1 0 0 1 1 1 Truth table for the logical AND operator ➢ Complementation:

The complementation operation is a unary operation, which is defined on a single variable. The symbol "-"

or " ' "

is normally used for complementation operator. It is also known as "NOT" operator. We can write, A' meaning "Take the complement of A", or (A+B)', meaning take the complement of A+B". The complementation of a variable is the reverse of its value. Thus, if A=0 then A' = 1and if A=1 then A'= 0. Input A Output A 0 1 1 0 Truth table for the logical NOT operator

Operator Precedence

The procedure of Boolean operators is as follows:- ➢ The expression is scanned from left to right. ➢ Expressions enclosed within parenthesis are evaluated first. ➢ All compliment (NOT) operations are performed next. ➢ All "." (AND) operations are performed after that. ➢ Finally all "+" (OR) operations are performed in the end. 6.9.2

Postulates of Boolean Algebra Postulate 1: (a) A=0 if and only if A is not equal to 1 (b) A=1 if and only if A is not equal to 0 Postulate 2: (a) x + 0 = x (b) x.1 = x Postulate 3: Commutative Law (a) x + y = y + x (b) x.y = y.x Postulate 4: Associative Law (a) x + (y + z) = (x + y) + z (b) x.(y.z) = (x.y).z Postulate 5: Distributive Law (a) x.(y + z) = x.y + x.z (b) x + y.z = (x + y).(x + z) Postulate 6: (a) x + x' = 1 (b) x.x' = 0 These postulates are the basic axioms of the algebraic structure, which need no proof. They are used to prove the theorems of Boolean Algebra. 6.9.3

Minimization of Boolean Functions When a Boolean function is implemented with logic gates, each literal in the function designates an input to a gate, and each termis implemented with a gate. Hence for a given function, the minimization of the number of literals, and the number of terms, will result in a circuit with fewer components. 6.9.4

Simplification of Boolean equations Example - 1 Given A = True, B = False, C = False (i) Solve X = (A.B)'.C = (T.F)'.F = F'.F = T.F = F (ii) Solve Y = (A'.B'.C)' = (T'.F'.F)' = (F.T.T)' = F' = T Given A = True, B = False, C = False (iii) Solve X = A.B'.C' + ((B.A')' + C.A)' = T.T'.F' + ((T.T')' + F.T)' = T.F.T + ((T.F)' + F.T)' = F + (F' + F)' = F + (T + F)' = F + T' = F + F = F (iv) Simplify Y =

A.B' + A.B = A.(B' + B) = A.T (X + X' = 1) = A (v) Simplify Y = A.B.C + A.B.C' + A.B' = A.B.(C + C') + A.B' = A.B + A.B' (X + X' = 1) = A.(B + B') = A (

X + X' = 1) 6.9.5 Karnaugh map (K-map) A Karnaugh map (K-map) is a pictorial method used to minimize Boolean expressions without having to use Boolean algebra theorems and equation manipulations. AK-map can be thought of as a special version of a truth table. Using a K-map, expressions with two to four variables are easily minimized. Expressions with five to six variables are more difficult but achievable, and expressions withseven or more variables are extremely difficult (if not impossible) to minimize using a K-map. Boolean algebra can be awkward in order to simplify expressions. Apart from being laborious (and requiring the remembering all the laws) the method can lead to solutions which, though they appear minimal, are not. The Karnaugh map provides a simple and straight-forward method of minimizing Boolean expressions. With the Karnaugh map Boolean expressions having up to four and even six variables can be simplified. The diagram below illustrates the correspondence between the Karnaugh map and the truth table for the general case of a two-variable problem.

The values inside the squares are copied from the output column of the truth table, therefore there is one square in the map for every row in the truth table. Around the edge of the Karnaugh map are the values of the two input variable. Ais along the top and B is down the left hand side. The diagram below explains this: The values around the edge of the map can be thought of as coordinates. So as an example, the square on the top right hand corner of the map in the above diagram has coordinates A=1 and B=0. This square corresponds to the row in the truth table where A=1 and B=0 and F=1. Note that the value in the F column represents a particular function to which the Karnaugh map corresponds. Example 1: Consider the following map. The function plotted is: Z = f(A,B) = A.B' + AB ➢ Note that values of the input variables form the rows and columns. That is the logic values of the variables A and B (with one denoting true form and zero denoting false form) form the head of the rows and columns respectively. ➢ Bear in mind that the above map is a one dimensional type which can be used to simplify an expression in two variables. ➢ There is a two-dimensional map that can be used for up to four variables, and a three-dimensional map for up to six variables. Using algebraic simplification, Z = A.B' + AB Z = A(B' + B)

Z = A Variable B becomes redundant. Referring to the map above, the two adjacent 1's are grouped together. Through inspection it can be seen that variable B has its true and false form within the group. This eliminates variable B leaving only variable A which only has its true form. The minimised answer therefore is Z = A. Example 2: Consider the expression Z = f(A,B) = A'.B' + A.B' +A'.B plotted on the Karnaugh map: Pairs of 1's are grouped as shown above, and the simplified answer is obtained by using the following steps: Note that two groups can be formed for the example given above, bearing in mind that the largest rectangular clusters that can be made consist of two 1s. Notice that a 1 can belong to more than one group. The first group labeled I, consists of two 1s which correspond to A = 0, B = 0 and A = 1, B = 0. Put in another way, all squares in this example that correspond to the area of the map where B = 0 contains 1s, independent of the value of A. So when B = 0 the output is 1. The expression of the output will contain the term B'. For group labeled II corresponds to the area of the map where A = 0. The group can therefore be defined as A'. This implies that when A = 0 the output is 1. The output is therefore 1 whenever B = 0 and A = 0. Hence the simplified answer is Z = A' + B' Summary Data stored in a coded form is basically of three types - Numeric, Alphabetic and Alphanumeric. Every computer stores these data types in a coded form i.e. in binary other number system. Number system are of two types - Non-Positional Number System and Positional Number System. Almost all computers uses binary numbers.

Information is handled in a

off(0). All information are

floating-point representation of a number needs two parts. The first part represents a signed, fixed-point number called the mantissa. The second part designates the position of the decimal (or binary) point is called the exponent. Boolean algebra is an

which are used by

Glossary Base The total number of digits used in a positional number system. It is also known as radix.
Binary
A
characteristic or property involving a selection, choice or condition in which there are two possibilities.
In number system, it refers to the system in which the base used is two, each number expressed in powers of two by using only two digits i.e. 0 and 1. Bit Acronym for Binary Digit i.e. 0 and 1. It stands for one binary piece of information.
Boolean Variable
A variable used in Boolean algebra. It can assume a value true or false.
Byte
A fixed number of adjacent bits, which represent a particular character or symbol. Normally a byte consists of eight bits.
Character A single alphabet numeric or special symbol that is used to represent data.
Decimal number A number system with a base of 10. The ten allowable digits System are 0,1,2,3,4,5,6,7,8,9.
Floating-point Signed numbers held in a fraction-exponent format. number Hexadecimal
A number system with a base of 16, Its digits range from 0 to
f. Number system Karnaugh map A Karnaugh map (K-map) is a pictorial method used to minimize Boolean expressions without having to use Boolean algebra
theorems and equation manipulations. Octal Number
A number system with a base of 8. The octal digits range from 0
System to 7. Truth Table
A table, which gives the output values for the various
input combinations in case of a logical expressions. Further Readings The following books are suggested for further reading: 1. Computer Fundamentals, P.K. Sinha, BPB publication 2. Computer System Arcitecture, M.Morris Mano, PHI 3. Digital Principles and applications, Malvino,Leach, TMH
Questions 1. Why binary number system is important in representing data in computer? Explain. 2. Explain the significance of base of a

number system. 3. What is the difference between positional and non-positional number systems? Give examples of both types of number systems. 4.

Find the decimal equivalent of the following binary numbers: a. 101100001.1011 b. 11011101 c. 1011.1011 d. 1101011 5.

Convert the decimal numbers into binary: a. 246.125 b. 765.525 c. 435 d. 1001 6. Convert the following numbers to decimal numbers: a. (B2CA)16 b. (110011)2 c. (2517)8 d. (152F)16 7. Convert

octal numbers to Hexadecimal number system: a.
6337 b. 6715 c. 7432 d. 4372 8.
What do you mean by mantissa and exponent? Explain. 9. What are the basic postulates of Boolean algebra? 10. Simplify the given Boolean equation Y =

A'.B'.C' + A.B'.C'+A'.B.C'+A.B.C'

Unit 7: Study of Logic Gates 7.0 Objectives 7.1 Introduction 7.2 Logic Gates 7.2.1 Inverter (NOT gate) 7.2.2 AND Gate 7.2.3 OR Gate 7.2.4 NAND Gate 7.2.5 NOR Gate 7.2.6 Exclusive-OR (XOR) and Exclusive NOR (XNOR) Gates 7.3 Equivalent Circuits 7.4 CODES 7.4.1 Character repertoire 7.4.2 Character Code 7.4.3 Character Encoding 7.4.4 ASCII Code 7.4.5 EBCDIC 7.4.6 Gray Code 7.4.7 Unicode 7.4.8 Binary Codes 7.5 Encoding and Decoding 7.0
Objectives After going through this unit you will be able to: ➢
Understand about
various logic gates like NOT, OR, AND, NAND, NOR, XOR, XNOR and how these gates perform the various operations on the given binary values at an instance. ➢ Understand about the coding system and definitions according to the systems requirement and how these codes are useful in the data representation. 7.1 Introduction Logic gates are useful to define a digital circuit and its implementation on a system. A digital circuit
having one or more inputs and one output signal called a gate. This unit will help you to understand the behavior of logic gates when the values are taken in input and generation of digital circuits using gates. 7.2 Logic Gates
Binary information is represented in digital computers by physical quantities called signals. Electrical signals such as voltage exist throughout
the computer in either one of two recognizable states. The two states represent a binary variable that can be
equal to 1 or 0. For example a particular digital computer may employ a signal of 3 volts to represent binary 1 and 0.5 volt to represent binary 0. The input terminals of digital circuits accept binary signals of 3 and 0.5 volts and the circuits respond at the output terminals with signals of 3 and 0.5 volts to represent binary input and output corresponding to 1 and 0 respectively. Binary logic deals with binary variables and
with binary operations that assume a logical meaning. It is used to describe in algebraic or tabular form, the manipulation and processing of binary information. The manipulation of binary information is done by logic circuits called Logic Gates. Gates are blocks of hardware that produce signals of binary 1 or 0 when input logic requirements are satisfied. Different types of
logic gates are used in digital systems. Each gate has a distinct graphical symbol and its

operations are described by the algebraic expression. The input and output operations and relationships are represented by the truth tables. Logic gates process signals, which represent true or false. Normally the positive supply voltage +Vs represent true and 0V represents false. Other terms, which are used, for the true and false states, are shown in the table on the right. It is best to be familiar with them all. A logic gate performs a logical operation on one or more logic inputs and produces a single logic output. The logic normally performed is Boolean logic and is most commonly found in digital circuits. Logic gates are primarily implemented electronically using diodes or transistors, but can also be constructed using electromagnetic relays, fluidics, optics, or even mechanical elements. Note: NAND and NOR are universal logic gates. 7.2.1 Inverter (NOT gate) The inverter (NOT circuit) performs a basic logic function called inversion or complementation. The purpose of the inverter is to change one logic level (HIGH / LOW) to the opposite logic level. In terms of bits, it changes a '1' to a '0' and vice versa. Following Figure shows the standard logic symbol for the inverter. Figure: Standard Logic Symbol for Inverter We generally express the logical operation of a gate with a truth table which lists all input combinations and the corresponding outputs. The truth table for the NOT gate is shown in the Table. INPUT OUTPUT A 0 1 1 0 The total number of possible input combinations (N) is determined by the mathematical formula: $N = 2^n$ Where n is the number of input variables. 7.2.2 AND Gate The AND gate, which is composed of two or more inputs and a single output, performs logical multiplication. The standard symbol for the AND gate is shown in Figure and its truth table listed in Table. The logical operation of the AND gate is such that the output is HIGH (1) when all the inputs are HIGH, otherwise it is LOW (0). Figure: Standard Logic Symbol for AND gate INPUT OUTPUT A B 0 0 0 0 1 0 1 0 0 1 1 1 7.2.3 OR Gate The OR gate, which is composed of two or more inputs and a single output, performs logical addition. The standard symbol for the OR gate is shown in Figure and its truth table listed in Table. The logical operation of the OR gate is such that the output is HIGH (1) when any of the inputs are HIGH, otherwise it is LOW (0). Figure: Standard Logic Symbol for OR gate INPUT OUTPUT A B 0 0 0 0 1 1 1 0 1 1 1 1 7.2.4 NAND Gate The NAND, which is composed of two or more inputs and a single output, is a very popular logic element because it may be used as a universal function. That is, it may be employed to construct an inverter, an AND gate, an OR gate, or any combination of theses functions. The term NAND is formed by the concatenation NOT-AND and implies an AND function with an inverted output. The standard symbol for the NAND gate is shown in Figure and its truth table listed in Table. The logical operation of the NAND gate is such that the output is LOW (0) only when all the inputs are HIGH (1).

Figure: Standard logic symbol for NAND gate INPUT OUTPUT A B 0 0 1 0 1 1 1 0 1 1 1 0 7.2.5 NOR Gate The NOR gate, which is composed of two or more inputs and a single output, also has a universal property. The term NOR is formed by the concatenation NOT-OR and implies an OR function with an inverted output. The standard symbol for the NOR gate is shown in Figure and its truth table listed in Table. The logical operation of the NOR gate is such that the output is HIGH (1) only when all the inputs are LOW. Figure: Standard logic symbol for NOR gate

INPUT OUTPUT A B 0 0 1 0 1 0 1 0 0 1 1 0 Truth table for NOR gate 7.2.6 Exclusive-OR (XOR) and Exclusive NOR (XNOR) Gates These gates are usually formed from the combination of the other logic gates already discussed. However, because of their functional importance, these gates are treated as basic gates with their own unique symbols. The truth tables for the XOR and XNOR gates, shown in Figure, are listed in Table. The Exclusive-OR is an "inequality" function and the output is HIGH (1) when the inputs are not equal to each other. Conversely, the Exclusive-NOR is an "equality" function and the output is HIGH (0) when the inputs are equal to each other. Figure: Standard logic symbols for: (a) XOR (b) XNOR INPUT XOR OUTPUT XNOR OUTPUT A B 0 0 0 1 0 1 1 0 1 0 1 0 1 1 0 1 Truth table for XOR and XNOR logic gates

7.3

Equivalent Circuits NAND and NOR are universal logic gates. Using DeMorgan's theorem and the Rules and laws of Boolean algebra proving this should be an easy task. Figure shows the equivalency between the basic logic gates and their NAND logic circuits counterpart. Similarly, Figure shows the equivalency between the basic logic gates and their NOR logic circuits counterpart. Figure: NAND Equivalent Circuits Note that i.e the complement of "A NAND B"
Figure: NOR Equivalent Circuits 7.4 Codes Any data to be accepted by computer is to be converted into binary form. Various codes representation of data are discussed in this section. 7.4.1 Character repertoire Character repertoire is a set of distinct characters. No specific internal presentation in computers or data transfer is assumed. The repertoire does not even define an ordering for the characters; ordering for sorting and other purposes is to be specified separately. A character repertoire is usually defined by specifying names of characters and a sample (or reference) presentation of characters in visible form.

00 Notice that a character repertoire may contain characters which look the same in some presentations but are regarded as logically distinct, such as Latin uppercase A, Cyrillic uppercase A, and Greek uppercase alpha. 7.4.2 Character Code A mapping, often presented in tabular form, which defines a one-to-one correspondence between characters in a character repertoire and a set of nonnegative integers. That is, it assigns a unique numerical code, a code position, to each character in the repertoire. In addition to being often pre- sented as one or more tables, the code as a whole can be regarded as a single table and the code positions as indexes. As synonyms for "code position", the following terms are also in use: code

number,

code value, code element, code point, code set value - and just code. Note: The set of nonnegative integers corresponding to characters need not consist of consecutive numbers; in fact, most character codes have "holes", such as code positions reserved for control functions or for eventual future. 7.4.3 Character Encoding A method or an algorithm for presenting characters in digital form by mapping sequences of code numbers of characters into sequences of octets. In the simplest case, each character is mapped to an integer in the range 0 - 255 according to a character code and these are used as such as octets. Naturally, this only works for character repertoires with at most 256 characters. For larger sets, more complicated encoding are needed. Encoding have names, which can be registered. Note: A character code assumes or implicitly defines a character repertoire. A character encoding could, in principle, be viewed purely as a method of mapping a sequence of integers to a sequence of octets. However, quite often an encoding is specified in terms of a character code (and the implied character repertoire).

The logical

structure is still the following: 1. A character repertoire specifies a collection of characters, such as "a", "!", and "ä". 2. A character code defines numeric codes for characters in a repertoire. For example, in the ISO 10646 character code the numeric codes for "a", "!", "ä", and "‰" (per mille sign) are 97, 33, 228, and 8240. (Note: Especially the per mille sign, presenting 0 /as a single character, can be shown incorrectly on display or on paper. That would be an illustration of the symptoms of the problems we are discussing.) 3. A character encoding defines how sequences of numeric codes are presented as (i.e., mapped

to) sequences of octets. In one possible encoding for ISO 10646, the string a!ä‰ is presented as the following sequence of octets (using two octets for each

character): 0, 97, 0, 33, 0, 228, 32, 48. 7.4.4

ASCII Code Most programming languages have a means of defining a character as a numeric code and conversely, converting the code back to the character. ASCII - American Standard Code for Information Interchange. A coding standard for characters, numbers, and symbols that is the same as the first 128 characters of the ASCII character set but differs from the remaining characters. ASCII is the American Standard Code for Information Interchange, also known as ANSI X3.4. There are many variants of this standard, typically to allow different code pages for language encoding, but they all basically follow the same format. ASCII is quite elegant in the way it represents characters, and

it is very easy to write code to manipulate upper/lowercase and check for valid data ranges.

ASCII is essentially a 7-bit code which allows the 8th most significant bit (MSB) to be used for error checking, however most modern computer systems tend to use ASCII values of 128 and above for extended character sets. The ASCII character set (excluding the extended characters defined by IBM) is divided into four groups of 32 characters. The definition of ASCII also specifies a set of control codes ("control characters") such as linefeed (LF) and escape (ESC). But the character repertoire proper, consisting of the printable characters of ASCII, is the following (where the first item is the blank, or space, character): ! " # $ % & ' ( ) * +, -. / 0 1 2 3 4 5 6 7 8 9 : ; > = < ? @

63%	MATCHING BLOCK 131/158	W

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z [\ ] ^ _ ' a b c d e f g h i j k l m n o p q r s t u v w x y z { | } ~

The first 32 characters, ASCII codes 0 through 1Fh, form a special set of non-printing characters called the control characters. We call them control characters because they perform various printer/ display control operations rather than displaying symbols.

Examples of common control characters include: ▪ carriage return (ASCII code 0Dh), which positions the cursor to the left side of the current line of characters, ▪ line feed (ASCII code 0Ah), which moves the cursor down one line on the output device ▪ back space (ASCII code 08h), which moves the cursor back one position to the left The second group of 32 ASCII character codes comprises various punctuation symbols, special

characters,

and the numeric digits. The most notable characters in this group include the: ▪ space character (ASCII code 20h) ▪ numeric digits 0 through 9 (ASCII codes 30h through 39h) Note that the numeric digits differ from their numeric values onlyin the high order nibble. By subtracting 30h from the ASCII code for any particular digit you can obtain the numeric equivalent of that digit. The third group of 32 ASCII characters is reserved for the upper case alphabetic characters. The ASCII codes for the characters "A" through "Z" lie in the range 41h through 5Ah. Since there are only 26 different alphabetic characters, the remaining six codes hold various special symbols. The fourth, and final, group of 32 ASCII character codes is reserved for the lower case alphabetic symbols, five additional special symbols, and another control character (delete). Note that the lower case character symbols use the ASCII codes 61h through 7Ah. If you compare the ASCII codes for the upper and lower case characters to binary, you will notice that the upper case symbols differ from their lower case equivalents in exactly one bit position. The only place these two codes differ is in bit five. Upper case characters always contain a zero in bit five; lower case alphabetic characters always contain a one in bit five. You can use this fact to quickly convert between upper and lower case. If you have an upper case character you can force it to lower case by setting bit five to one. If you have a lower case character and you wish to force it to upper case, you can do so by setting bit five to zero. You can toggle an alphabetic character between upper and lower case by simply inverting bit five. ASCII Code Chart Indeed, bits five and six determine which of the four groups in the ASCII character set you're in: Bit 6 Bit 5 Group 0 0 Control Characters

0 1 Digits and Punctuation 1 0 Upper Case and Special 1 1 Lower Case and Special So, you could, for instance, convert any upper or lower case (or corresponding special) character to its equivalent control character by setting bits five and six to zero. The ASCII codes of the numeric digit characters: Char Dec Hex '0' 48 30 '1' 49 31 '2' 50 32 '3' 51 33 '4' 52 34 '5' 53 35 '6' 54 36 '7' 55 37 '8' 56 38 '9' 57 39 The decimal representations of these ASCII codes are not very enlightening. However, the hexadecimal representation of these ASCII codes reveals something very important; the low order nibble of the ASCII code is the binary equivalent of the represented number. Bit seven in standard ASCII is always zero. This means that the ASCII character set consumes only half of the possible character codes in eight bits or a byte. The PC uses the remaining 128 character codes for various special characters including international characters (those with accents, etc.), math symbols, and line drawing characters. Note that these extra characters are a non-standard extension to the ASCII character set. Most printers support the PC's extended character set. Despite some major shortcomings, ASCII data is the standard for data interchange across computer systems and programs. Most programs can accept ASCII data; likewise most programs can produce

ASCII data. If you will program in the assembly language you will be dealing with ASCII characters, and it would for

that reason be wise to study the layout of the character set and memorize a few

key ASCII

codes (e.g., "0", "A", "a", etc.). ASCII Table Format X 3 X 2 X 1 X 0 X 6 X 5 X 4 010 011 100 101 110 111 0000 SP 0 @ P P 0001 ! 1 A Q a q 0010 " 2 B R b r 0011 # 3 C S C S 0100 $ 4 D T d t 0101 % 5 E U e u 0110 & 6 F V f v 0111 ' 7 G W g w 1000 (8 H X h x 1001) 9 I Y i y 1010 * : J Z j z 1011 + ; K k 1100 , > L l 1101 - = M m 1110 . < N n 1111 / ? O o

7.4.5

EBCDIC EBCDIC (Extended Binary Coded Decimal Interchange Code) is a character encoding set used by IBM mainframes. EBCDIC uses the full 8 bits available to it, so parity checking cannot be used on an 8 bit system. Also, EBCDIC has a wider range of control characters than ASCII. The character encoding is based on Binary Coded Decimal (BCD), so the contiguous characters in the alphanumeric range are formed up in blocks of up to 10 from 0000 binary to 1001 binary. Non alphanumeric characters are almost all outside the BCD range. There are four main blocks in the EBCDIC code page: 0000 0000 to 0011 1111 is reserved for control characters; 0100 0000 to 0111 1111 are for punctuation; 1000 0000 to 1011 1111 for lowercase characters and 1100 0000 to 1111 1111 for uppercase characters and numbers. EBCDIC is mostly used

in IBM mainframe and midrange systems. 7.4.6

Gray Code Gray coding is an important code and is used for its speed, it is also relatively free from errors. In pure binary coding or 8421 BCD then counting from 7 (0111) to 8 (1000) requires 4 bits to be changed simultaneously. If this does not happen then various numbers could be momentarily generated during the transition so creating spurious numbers, which could be read. Gray coding avoids this since only one-bit changes between subsequent numbers. To construct the code there are two simple rules. First start with all 0s and then proceed by changing the least significant bit (LSB) which will bring about a new state. Gray Code Table Decimal Gray Code Binary 0 0000 0000 1 0001 0001 2 0011 0010 3 0010 0011 4 0110 0100 5 0111 0101 6 0101 0110 7 0100 0111 8 1100 1000 9 1101 1001 10 1111 1010 11 1110 1011 12 1010 1100 13 1011 1101 14 1001 1110 15 1000 1111 ... ... The first 16 Gray coded numbers are indicated below. Decimal Gray Code 0 0000 1 0001 2 0011

3 0010 4 0110 5 0111 6 0101 7 0100 8 1100 9 1101 10 1111 11 1110 12 1010 13 1011 14 1001 15 1000 To convert a Gray-coded number to binary then follow this method: 1. The binary number and the Gray-coded number will have the same number of bits 2. The binary MSB (left-hand bit) and Gray code MSB will always be the same 3. To get the binary next-to-MSB (i.e. next digit to the right) add the binary MSB and the gray code next-to-MSB. Record the sum, ignoring any carry. 4. Continue in this manner right through to the end. Gray coding is a non-BCD, non-weighted reflected binary code. 7.4.7

Unicode Unicode is a standard, by the Unicode Consortium, which defines a character repertoire and character code intended to be fully compatible with ISO 10646, and an encoding for it. ISO 10646 is more general (abstract) in nature, whereas Unicode "imposes additional constraints on implementations to ensure that they treat characters uniformly across platforms and applications". Unicode is an entirely new idea in setting up binary codes for text or script characters. Officially called the Unicode Worldwide Character Standard, it is a system for "the interchange, processing, and display of the written texts of the diverse languages of the modern world." It also supports

many classical and

historical texts in a number of languages. Currently, the Unicode standard contains 34,168 distinct coded characters derived from 24 supported language scripts. These characters cover the principal written languages of the world. Unicode was originally designed to be a 16-bit code, but it was extended so that currently code positions are expressed as integers in the hexadecimal range 0..10FFFF (decimal 0..1 114 111). That space is divided into 16-bit "planes". Until recently, the use of Unicode has mostly been limited to "Basic Multilingual Plane (BMP)" consisting of the range 0..FFFF. The ISO 10646 and Unicode character repertoire can be regarded as a superset of most character repertoires in use. However, the code positions of characters vary from one character code to another. Unicode defines two mapping methods: ➢ UTF (Unicode Transformation Format) encoding ➢ UCS (Universal Character Set) encoding The encoding includes: ➢ UTF-7 — a relatively unpopular 7-bit encoding, often considered obsolete (not part of The Unicode Standard but rather an RFC) ➢ UTF-8 — an 8-bit, variable-width encoding, which maximizes compatibility with ASCII. ➢ UTF-EBCDIC — an 8-bit variable-width encoding, which maximizes compatibility with EBCDIC. (not part of The Unicode Standard) ➢ UCS-2 — a 16-bit, fixed-width encoding that only supports the BMP, considered obsolete. (not part of The Unicode Standard) ➢ UTF-16 — a 16-bit, variable-width encoding ➢ UTF-32 — a 32-bit, fixed-width encoding ➢ UCS-4 — functionally equivalent to UTF-32, but not part of The Unicode Standard. The numbers in the names of the encoding indicate the number of bits in one code value (for UTF encoding) or the number of bytes per code value (for UCS) encoding. UTF-8 and UTF-16 are probably the most commonly used encoding. 7.4.8

Binary Codes The usual way of expressing a decimal number in terms of a binary number is known as pure binary coding. A number of other techniques can be used to represent a decimal number. These are summarized below. 8421 BCD Code In the 8421 Binary Coded Decimal (BCD) representation each decimal digit is converted to its 4-bit pure binary equivalent. For example: 57 dec = 0101 0111 bcd

Addition is analogous to decimal addition with normal binary addition taking place from right to left. For example, 6 0110 BCD for 6 42 0100 0010 BCD for 42 +3 0011 _ 1001 BCD for 3 BCD for 9 +27 0010 0111 _ 0110 1001 BCD for 27 BCD for 69 Where the result of any addition exceeds 9(1001) then six (0110) must be added to the sum to account for the six invalid BCD codes that are available with a 4-bit number. This is illustrated in the example below 8 1001 BCD for 8 +7 0111 _ 1111 BCD for 7 exceeds 9 (1001) so 0110 add six (0110) 0001 0101 BCD for 15 Note that in the last example the 1 that carried forward from the first group of 4 bits has made a new 4- bit number and so represents the "1" in "15". In the examples above the BCD numbers are split at every 4-bit boundary to make reading them easier. This is not necessary when writing a BCD number down. This coding is an example of a binary coded (each decimal number maps to four bits) weighted (each bit represents a number, 1, 2, 4, etc.) code. 4221 BCD Code The 4221 BCD code is another binary coded decimal code where 4, 2, 2 and 1 weights each bit respectively. Unlike BCD coding there are no invalid representations. The decimal numbers 0 to 9 have the following 4221 equivalents: Decimal 4221 1's complement 0 0000 1111 1 0001 1110 2 0010 1101 3 0011 1100 4 1000 0111 5 0111 1000 6 1100 0011 7 1101 0010 8 1110 0001 9 1111 0000 the 1's complement of a 4221 representation is important in decimal arithmetic. In forming the code remember the following rules: ➢ Below decimal 5 use the right-most bit representing 2 first ➢ Above decimal 5 use the left-most bit representing 2 first ➢ Decimal 5 = 2+2+1 and not 4+1 7.5

ENCODING AND DECODING EBCDIC to ASCII and ASCII to EBCDIC Conversion The main reason to convert an EBCDIC file to an ASCII file is so that the file can be used on a PC, a server, or some UNIX-based system. Keep these points in mind when converting mainframe files. ➢ You may not need to write a conversion program at all, if your record contains all text data. Most mainframe-based file transfer programs

have

an option to automatically convert a file from EBCDIC to ASCII while performing the download. ➢ Often, however, you will not want to convert the whole record. If the record contains any non-text fields (such as binary numeric fields, packed numeric fields, hex or packed date fields, or bit fields), the bytes in those fields cannot simply be "translated" from EBCDIC to ASCII. They must either be left "as is" (if the PC system can handle that type of data) or reformatted for use on the PC. If your records do contain such non-text data, then you can not use the file transfer program to perform the conversion. ➢ In addition to converting text from EBCDIC to ASCII, any binary, packed or BCD numeric fields may need to be reformatted to text-type numeric fields for the PC.

➢ You may also need to convert dates from packed, hex, or BCD values in the mainframe file to text- type date fields for the PC. ➢ In addition to translating the data to ASCII, the data may be much easier to use on the PC if you delimit the fields with commas. Your PC program will probably also expect to see quotation marks around the character fields (especially if they contain embedded spaces or commas). The PC program may also require dates to have slashes in them, which dates in mainframe records usually do not have. Table: ASCII and EBCDIC Characters Character Code ASCII EBCDIC Control/ To Control/ To Decimal Octal Hex Graphic EBCDIC Hollerith Graphic ASCII (Octal) (Octal) 0 000 00 NUL 000 12 0 1 8 9 NUL 000 1 001 01 SOH 001 12 1 9 SOH 001 2 002 02 STX 002 12 2 9 STX 002 3 003 03 ETX 003 12 3 9 ETX 003 4 004 04 EOT 067 7 9 PF 234 5 005 05 ENQ 055 0 5 8 9 HT 011 6 006 06 ACK 056 0 6 8 9 LC 206 7 007 07 BEL 057 0 7 8 9 DEL 177 8 010 08 BS 026 11 6 9 GE 227 9 011 09 HT 005 12 5 9 RLF 215 10 012 0A LF 045 0 5 9 SMM 216 11 013 0B VT 013 12 3 8 9 VT 013 12 014 0C FF 014 12 4 8 9 FF 014 13 015 0D CR 015 12 5 8 9 CR 015 14 016 0E SO 016 12 6 8 9 SO 016 15 017 0F SI 017 12 7 8 9 SI 017 16 020 10 DLE 020 12 11 1 8 9 DLE 020 17 021 11 DC1 021 11 1 9 DC1 021 18 022 12 DC2 022 11 2 9 DC2 022 19 023 13 DC3 023 11 3 9 TM 023 20 024 14 DC4 074 4 8 9 AES 235 21 025 15 NAK 075 5 8 9 NL 205 22 026 16 SYN 062 2 9 BS 010

23 027 17 ETB 046 0 6 9 IL 207 24 030 18 CAN 030 11 8 9 CAN 030 25 031 19 EM 031 11 1 8 9 EM 031 26 032 1A SUB 077 7 8 9 CC 222 27 033 1B ESC 047 0 7 9 CU1 217 28 034 1C FS 034 11 4 8 9 IFS 034 29 035 1D GS 035 11 5 8 9 IGS 035 30 036 1E RS 036 11 6 8 9 IRS 036 31 037 1F US 037 11 7 8 9 IUS 037 32 040 20 SP space 100 Blank DS 200 33 041 21 ! 117 12 7 8 SOS 201 34 042 22 " 177 7 8 FS 202 35 043 23 # 173 3 8 203 36 044 24 $ 133 11 3 8 BYP 204 37 045 25 % 154 0 4 8 LF 012 38 046 26 & 120 1 2 ETB 027 39 047 27 ' apost. 175 5 8 ESC 033 40 050 28 (115 12 5 8 210 41 051 29) 135 11 5 8 211 42 052 2A * 134 11 4 8 SM 212 43 053 2B + 116 12 6 8 CU2 213 44 054 2C , comma 153 0 3 8 214 45 055 2D - hyphen 140 11 ENQ 005 46 056 2E . period 113 12 3 8 ACK 006 47 057 2F / 141 0 1 BEL 007 48 060 30 0 360 0 220 49 061 31 1 361 1 221 50 062 32 2 362 2 SYN 026 51 063 33 3 363 3 223 52 064 34 4 364 4 PN 224 53 065 35 5 365 5 RS 225 54 066 36 6 366 6 UC 226 55 067 37 7 367 7 EOT 004 56 070 38 8 370 8 230 57 071 39 9 371 9 231 58 072 3

A : 172 2 8 232 59 073 3B ; 136 11 6 8 CU3 233 60 074 3C > 114 12 4 8 DC4 024 61 075 3D = 176 6 8 NAK 025 62 076 3E < 156 0 6 8 023 63 077 3F ? 157 0 7 8 SUB 032 64 100 40 @ 174 48 SP 040 65 101 41 A 301 12 1 240 66 102 42 B 302 12 2 241 67 103 43 C 303 12 3 242 68 104 44 D 304 12 4 243 69 105 45 E 305 12 5 244 70 106 46 F 306 12 6 245 71 107 47 G 307 12 7 246 72 110 48 H 310 12 8 247 73 111 49 I 311 12 9 250 74 112 4A J 321 11 1 ¢ 133 75 113 4B K 322 11 2 . period 056 76 114 4C L 323 11 3 > 074 77 115 4D

M 324 11 4 (050 78 116 4E

N 325 11 5 + 053 79 117 4F

O 326 11 6 | 041 80 120 50

underline 155 0 5 8 136 96 140 60 ` grave 171 1 8 - hyphen 055 97 141 61 a 201 12 0 1 / 057 98 142 62 b 202 12 0 2 262 99 143 63 c 203 12 0 3 263 100 144 64 d 204 12 0 4 264 101 145 65 e 205 12 0 5 | 265 102 146 66 f 206 12 0 6 266 103 147 67

g 207 12 0 7 267 104 150 68 h 210 12 0 8 270 105 151 69 i 211 12 0 9 | 271 106 152 6A j 221 12 11 1 | 174 107 153 6B k 222 12 11 2 , comma 054 108 154 6C l 223 12 11 3 % 045 109 155 6D

m 224 12 11 4 _unlin 137 110 156 6E

n 225 12 11 5 < 076 111 157 6F

o 226 12 11 6 ? 077 112 160 70

p 227 12 11 7 272 113 161 71 q 230 12 11 8 273 114 162 72 r 231 12 11 9 274 115 163 73 s 242 11 0 2 275 116 164 74 t 243 11 0 3 276 117 165 75 u 244 11 0 4 277 118 166 76 v 245 11 0 5 300 119 167 77 w 246 11 0 6 301 120 170 78 x 247 11 0 7 302 121 171 79 y 250 11 O 8 ' 140 122 172 7A z 251 11 0 9 : 072 123 173 7B { 300 12 0 # 043 124 174 7C | 152 12 11 @ 100 125 175 7D } 320 11 0 ' (apos) 047 126 176 7E - 241 11 0 1 = 075 127 177 7F DEL 007 12 7 9 " 042 128 200 80 040 11 0 1 303 8 9 129 201 81 041 0 1 9 a 141 130 202 82 042 0 2 9 b 142 131 203 83 043 0 3 9 c 143 132 204 84 044 0 4 9 d 144 133 205 85 025 11 5 9 e 145 134 206 86 006 12 6 9 f 146 135 207 87 027 11 7 9 g 147 136 210 88 050 0 8 9 h 150 137 211 89 051 0 1 8 9 i 151 138 212 8A 052 0 2 8 9 304 139 213 8B 053 0 3 8 9 305 140 214 8C 054 0 4 8 9 306 141 215 8D 011 12 1 8 9 307 142 216 8E 012 12 2 8 9 310 143 217 8F 033 11 3 8 9 311 144 220 90 060 12 11 0 1 8 9 312 145 221 91 061 1 9 j 152 146 222 92 032 11 2 89 k 153 147 223 93 063 3 9 l 154 148 224 94 064 4 9 m 155 149 225 95 065 5 9 n 156 150 226 96 066 6 9 o 157 151 227 97 ∅ 010 12 8 9

p 160 152 230 98 070 8 9 q 161 153 231 99 071 1 8 9 r 162 154 232 9

A 072 2 8 9 313 155 233 9B 073 3 8 9 314 156 234 9C 004 12 4 9 315 157 235 9D 024 11 4 9 316 158 236 9E 076 6 8 9 317 159 237 9F 341 11 0 1 9 320 160 240 A0 ∅ 101 12 0 1 9 321 161 241 A1 102 12 0 2 9 ~ 176 162 242 A2 103 12 0 3 9 s 163 163 243 A3 104 12 04 9 t 164 164 244 A4 105 12 0 5 9 u 165 165 245 A5 106 12 0 6 9 v 166 166 246 A6 107 12 0 79 w 167 167 247 A7 110 12 0 8 9 x 170 168 250 A8 111 12 1 8 y 171 169 251 A9 121 12 11 1 9 z 172 170 252 AA 122 12 11 2 9 322 171 253 AB 123 12 11 3 9 323 172 254 AC 124 12 11 4 9 324 173 255 AD 125 12 11 5 9 325 174 256 AE 126 12 11 6 9 326 175 257 AF 127 12 11 7 9 327 176 260 B0 130 12 11 8 9 330 177 261 B1 131 11 1 8 331 178 262 B2 142 11 0 2 9 332 179 263 B3 143 11 0 3 9 333 180 264 B4 144 11 0 4 9 334 181 265 B5 145 11 0 5 9 335 182 266 B6 146 11 0 6 9 336 183 267 B7 147 11 0 7 9 337 184 270 B8 ∅ 150 11 0 8 9 340 185 271 B9 151 0 1 8 341 186 272 BA 160 12 11 0 342 187 273

BB 161 12 11 0 1 9 343 188 274 BC 162 12 11 0 2 9 344 189 275 BD 163 12 11 0 3 9 345 190 276 BE 164 12 11 0 4 9 346 191 277 BF 165 12 11 0 5 9 347 192 300 C0 166 12 11 0 6 9 { 173 193 301 C1 167 12 11 0 7 9

A 101 194 302 C2 170 12 11 0 8 9 B 102 195 303 C3 200 12 0 1 8 C 103 196 304 C4 212 12 0 2 8 D 104 197 305 C5 213 12 0 3 8 E 105 198 306 C6 ∅ 214 12 0 4 8 F 106 199 307 C7 ∅ 215 12 0 5 8 G 107 200 310 C8 216 12 0 6 8 H 110 201 311 C9 217 12 0 7 8 I 111 202 312 CA 220 12 11 1 8 350 203 313 CB 232 12 11 2 8 351 204 314 CC 233 12 11 3 8 352 205 315 CD 234 12 11 4 8 353 206 316 CE 235 12 11 5 8 354 207 317 CF 236 12 11 6 8 355 208 320 D0 237 12 11 7 8 } 175 209 321 D1 240 11 0 1 8 J 112 210 322 D2 252 11 0 2 8 K 113 211 323 D3 253 11 0 3 8 L 114 212 324 D4 254 11 0 4 8 M 115 213 325 D5 255 11 0 5 8

N 116 214 326 D6 256 11 0 6 8 O 117 215 327 D7 257 11 0 7 8 P 120 216 330 D8 260 12 11 0 1 8 Q 121 217 331 D9 261 12 11 0 1 R 122 218 332 DA 262 12 11 0 2 356 219 333 D8 263 12 11 0 3 357 220 334 DC 264 12 11 0 4 360 221 335 DD 265 12 11 0 5 361 222 336 DE 266 12 11 0 6 362 223 337 DF 267 12 11 0 7 363 226 342 E2 272 12 11 0 2 8 S 123 227 343 E3 273 12 11 0 3 8 T 124 228 344 E4 274 12 11 0 4 8 U 125 229 345 E5 275 12 11 0 5 8 V 126 230 346 E6 276 12 11 0 6 8 W 127 231 347 E7 277 12 11 0 7 8 X 130 232 350 E8 312 12 0 2 8 9 Y 131 233 351 E9 313 12 0 3 8 9

Z 132 234 352

EA 314 12 0 4 8 9 364 235 353 EB 315 12 0 5 8 9 365 236 354 EC 316 12 0 6 8 9 366 237 355 ED 317 12 0 7 8 9 367 238 356 EE 332 12 11 2 8 9 370 239 357 EF 333 12 11 3 8 9 371 240 360

F0 334 12 11 4 8 9 0 060 241 361 F1 335 12 11 5 8 9 1 061 242 362 F2 336 12 11 6 8 9 2 062 243 363 F3 337 12 11 78 9 3 063 244 364 F4 352 11 0 2 8 9 4 064 245 365 F5 353 11 0 3 8 9 5 065 246 366 F6 354 11 0 4 8 9 6 066 247 367 F7 355 11 05 8 9 7 067 248 370 F8 356 11 0 6 8 9 8 070 249 371 F9 357 11 0 7 8 9 9 071 250 372

FA 372 12 11 0 2 8 9 372 251 373 FB 373 12 11 0 3 8 9 373 252 374 FC 374 12 11 0 4 8 9 374 253 375 FD 375 12 11 0 5 8 9 375 254 376 FE 376 12 11 0 6 8 9 376 255 377 FF 377 12 11 0 7 8 9 377 Summary Logic gates are useful to define a digital circuit and its implementation on a system. A digital circuit having one or more output signal called a gate. Binary logic deals with binary variables and

with binary operations that assume a logical meaning. It is used to describe in algebraic or tabular form, the manipulation and processing of binary information.

A logic gate performs a logical operation on one or more logic inputs and produces a single logic output. The logic normally performed is Boolean logic and is most commonly found in digital circuits. A digital circuit can be made by using the following gates like NOT, OR, AND, NAND, NOR, XOR, XNOR gates. CODES are used to define how data can be used by a digital system and in which form. Different coding mechanisms can be used in a system. ASCII - American Standard Code for Information Interchange. A coding standard for characters, numbers, and symbols that is the same as the first 128 characters of the ASCII character set but differs from the remaining characters.

EBCDIC (Extended Binary Coded Decimal Interchange Code) is a character encoding set used by IBM mainframes. EBCDIC uses the full 8 bits available to it, so parity checking cannot be used on an 8 bit system. Also, EBCDIC has a wider range of control characters than ASCII. Gray coding is an important code and is used for its speed, it is also relatively free from errors. In pure binary coding or 8421 BCD then counting from 7 (0111) to 8 (1000) requires 4 bits to be changed simultaneously. Unicode is an entirely new idea in setting up binary codes for text or script characters. Officially called the Unicode Worldwide Character Standard, it is a system for "the interchange, processing, and display of the written texts of the diverse languages of the modern world." It also supports many classical and historical texts in a number of languages. The usual way of expressing a decimal number in terms of a binary number is known as pure binary coding and is discussed in the Number Systems section. A number of other techniques can be used to represent a decimal number. Further Readings 1. Digital Principles and Applications by Leach and Malvino, Fifth Edition. 2. Digital Circuits by R.P. Jain 3. Digital Circuit Design by Salivahanan

Questions 1. Write an expression for the XOR and XNOR gate. Also define these gates with the appropriate logic diagrams and truth tables. 2. Write an expression for NAND and NOR gate with the suitable logic symbol and truth table. 3. Draw the logic diagrams for the followings: (a) Invert AND A′B′C′= (A+B+C)′ (b) Invert OR A′+B′+C′= (ABC)′ 4. Give some common control characters example. 5. What is Unicode? Define some encoding mechanisms used in Unicode. 6. Define 8421- BCD code with a suitable example. 7. Define the table of Gray code for first 16 numbers.

UNIT 8: Operating System Concepts 8.0 Objectives 8.1 Introduction 8.2 Layered View of a Computer System 8.3 Basic Functions of an Operating System 8.4 Classification of Operating Systems 8.5 Operating System Components 8.5.1 Process Management 8.5.2 Memory Management 8.5.3 File Management 8.5.4 I/O System Management 8.5.5 Secondary Memeory Management 8.5.6 Networking 8.5.7 Protection System 8.5.8 Command Interpreter System 8.6 Operating System Services 8.7 System Calls 8.8 System Programs 8.9 Kernel 8.10 Device Drivers 8.0 Objectives This chapter provides a general overview of ➢ Operating System components ➢ Operating System classification ➢ Functions of each component ➢ System Programs and Device Drivers 8.1 Introduction

The operating system is the most fundamental of all system programs, which hides the complexity of the bare hardware from the programmers or users. This can be defined as

92%	MATCHING BLOCK 135/158	W
a layer of software on top of hardware, to manage all parts of the		

computer system, and present the user with an interface or a virtual machine that is easier to understand and program. It controls all the computer resources (hard and soft parts)

and provides the base upon which the application program can be written

and executed. This form primarily interfaces between the user and machine and insulates the user from complex hardware details. Fig. 8.1 Operating System supporting various devices and applications. Generally software can be divided into two categories: ➢ System software: - Which manage the operation of computer itself. The system software is further divided into two categories: essential and desirable. Essential system software is necessary for the functioning of the machine. The most common example is the operating system. Desirable software are written for system uses but depends on the users need. Examples of desirable system software are compiler, interpreter, command interpreter, assembler, etc. ➢ Application program: - These are application specific and are developed to solve the problems for users. We have large number of such software such as MS Word, games, banking software, railway reservation system, etc. 8.2 Layered View of A Computer System The following diagram provides logical view of a typical computer system. In this computer system is divided into six layers. Banking system Adventure Games Research System WS Compiler Editors Command Interpreter Operating System Machine Language

Fig. 8.2 Layer diagram of a modern computer 1. Physical Device layer: This is the lowest layer, which consists of all hardware parts of the system such as ICs, Power Supplies, CRTs etc. 2. Micro program Level: Primitive software layer which directly controls the devices and provide clear interface to the next layer. At this level we have microinstruction to directly interact with hardware. This also provides logical view of internal CPU, data Path and either software controlled or hardwired controlled. 3. Machine language level: Typically consist of 50- 300 instruction in assembly language 4. Operating System Layer: Hides the complexity of hardware details and gives the programmer a more convenient set of instruction to work with. 5. Desirable Software layer: It consists of software such as command interpreter (shell), compilers, editors etc. 6. Application Program Layer: It consists of all kinds of application software required by the user. It includes word processor, spread sheets etc. 8.3 Basic Functions of an Operating System Act as an extended machine One of the primitive functions of OS is to

| 76% | **MATCHING BLOCK 136/158** | W |

present the user an extended machine or virtual machine that is easier to program and

understand by providing a variety of services that programs can obtain using special instructions called system calls. Act as a resource manager Resources are used to complete the assigned activity. Resource includes processors, memory, timer, disk, hardware etc. OS allocates and de-allocates the required resources to the processes as per the need and availability and manages the conflicts between processes for resource requests. OS completely controls the access of resources by managing time and space requirement of different programs (processes). Account of resource uses and their track are managed by OS. Protection and consistency of the system is also ensured by OS. Operating system provides Application programming interface (API) for the convenience of the user.

Micro program Physical devices

8.4

Classification of Operating Systems Operating systems can be classified into following categories: Main-Frame OS: The objective of the main-frame OS is to handle a large amount of computations and large number of I/O activities such as, 1000 disks and thousands of GB of data. Main frame OS works where large data processing capability required such as Web Servers, business transactions etc. Services of the main frames are provided in many modes like: Batch process (non-interactive mode), transaction process, time sharing etc. Server OS: Servers are large computer systems designed to support multiuser, multiprocessing and parallel processing similar to mainframes computer systems. They need special powerful operating system to support large number of users at the same time using time sharing concept and provides print service, file service, web service, DN service etc. are known as server OS. Example: UNIX, Linux, windows 2000, Novell are the examples of server operating systems. Multiprocessor OS: The computer systems having more than one processor or multiple CPUs need special operating system called multiprocessor OS. The services required in these systems are much more complex and different as this has to control more than one CPU. These include allocation of tasks to different CPUs. Real -time OS: A real-time system may be defined as one whose principal measure of performance is whether or not it meets prespecified task timing constraints or deadlines". These types of operating systems are used to control scientific devices and similar small instruments where memory and resources are crucial. These types of devices have very limited or no end user utilities, so more effort should go into making the OS really memory efficient and fast (less coding), so as to minimize the execution time, in turn saving on power as well. Hard real-time: • Have stringent timing requirements • Timing constraint of safety critical tasks must be satisfied under all circumstances • The failure of computer to meet certain task execution deadline can result in serious consequences. Soft real-time: • Have deadlines but not so stringent.

• Failure to meet soft deadlines will not result in hazardous situation. • The utility of results produced by a task with a soft deadline decreases over time after the deadline expires. Single User single Tasking: This is very small and simple operation system designed to control or process very small operation on small devices. The handhelds or the palmtop computers are good examples of this type of systems. Example: Windows mobile etc. Single User Multitasking: This is the most common type of operating system used for personal computers. These type of OS does not provide support for multiusers. Microsoft windows and Apple Macintosh are the examples of this type. These can perform Multitasking operations, like for example playing a multimedia file; downloading a file from the internet and editing a Text file simultaneously. Example: Windows Vista, Mac X tiger etc. Fig. 8.3 Typical look of Windows OS screen. Fig. 8.4 Typical look of a MAC OS screen. Multiuser: These types of operating systems provide support for multiple users to use the system resources simultaneously. This should not be confused with the multiuser accounts in windows or similar, the main difference being, the network administrator is the only actual user in Windows XP or equivalent or Macs and one more difference being that in OS like UNIX more than one user can simultaneously login and work while this is not possible in windows. Example: UNIX, Linux etc. Embedded OS: Small operating system designed for some specific application: - These OS are Limited in capabilities, power, function and size. Example: PDA (personal digital assistant), mobile devices, smart home appliances.

Smart card OS: Smart card operating systems are very small in size and function. ➢ They have very small processing power and memory. ➢ Some of them can handle a single function such as electronic payments ➢ Other can handle multiple functions on the same smart card such as college ID card may be used as attendance card, library card as well as debit card for college canteen. ➢ They are generally proprietary systems. ➢ Some smart cards are java oriented i.e. ROM on the smart card have JVM (Java Virtual Machine), java applets are downloaded to the card and are interpreted by JVM interpreter. Some of these cards can handle multiple java applets at the same time leading to multi programming support. ➢ Scheduling, resource management and protection are relevant issues, if card supports multiprogramming. 8.5 System Components An operating system contains following components: 8.5.1

73%	**MATCHING BLOCK 137/158**	W

Process Management A process is a program in execution, which needs certain resources, including CPU time, memory, files, and I/O devices,

to accomplish its task.
The operating system is responsible for the following activities in connection with
process management: •
Process creation and deletion. • Process suspension and resumption. • Provision of mechanisms for process synchronization and
process communication • Deadlock handling 8.5.2
Memory Management
Memory is
a large array of words or bytes,
each with
its own address.
It
is a collection of quickly accessible data shared by the CPU and
I/O devices.
Main memory is a volatile storage device. It loses its contents in the case of system failure or power off. Memory management is the part of
operating system which
is responsible for the many activities in connections with memory. It keeps track of which parts of memory are currently being used and
by whom. MMU decides which processes
to load when
memory space becomes available. Allocation and
deallocation of memory space is controlled by memory management unit of operating system. Fig. 8.8 Memory Hierarchy

Computer architectures arrange the computer's memory in a hierarchical manner, starting from the fastest registers, CPU cache, Random Access Memory and disk storage. An operating system's memory manager coordinates the use of these various types of memory by tracking which one is available, which is to be allocated or de allocated and how to move data between them. Virtual memory, increases the amount of memory available for each process by making the disk storage looks like main memory. There is a speed penalty associated with using disks or other slower storage as memory – if running processes require significantly more RAM than is available, the system may start thrashing (will be discussed later in memory management). This

can happen either because one process requires a large amount of RAM or because two or more processes compete for a larger amount of memory than

is

available. Another important part of memory management is managing virtual addresses. If multiple processes are in memory at once, they must be prevented from interfering with each other's memory. This is achieved by having separate address spaces. Each process sees the whole virtual address space, typically from address 0 up to the maximum size of virtual memory, as uniquely assigned to it.

Fig. 8.9 Data transfer from disk

to register The operating system maintains page table that match virtual addresses to physical addresses. These memory allocations are tracked so that when a process terminates, all memory

used by that process can be made available for other processes. The operating system can also write inactive memory pages to secondary storage. This process is called "paging" or "swapping". It is also typical for operating systems to employ otherwise unused physical memory as a page cache; requests for data from a slower device can be retained in memory to improve performance. The operating system can also pre-load the in-memory cache with data that may be requested by the user in the near future,

is called prepaging. 8.5.3

100%	**MATCHING BLOCK 138/158**	W

File Management A file is a collection of related information

defined by its creator.

Commonly, files represent programs (both source and object forms) and data.

The operating system is responsible for the following activities in connections with file management: •

File creation and deletion. •

Directory creation and deletion. • Support of

primitives for manipulating files and directories. • Mapping files onto secondary storage. •

File backup on stable (nonvolatile) storage media. OS keeps track of the location of various parts of the file on the device. It keeps the information about a file, its size, ownership, status, its use etc. All the necessary mechanisms to access and modify the file are kept by the OS. It enforces protection requirements in a multi-user System. Directory Structure In general Operating System supports following directory structures- ➢ Single directory: - it contains only one directory per user, i.e. no further subdirectories. ➢ Two level directory: - It allows subdirectories within a directory. ➢ Tree structured directory: - it allows many level directories within a directory. Path name of a directory can be given in two ways: ➢ Absolute path name

➢ Relative path name Working Directory on current directory

Fig. 8.10 Directory entry of each file 8.5.4 I/O System Management or Device management OS keeps track of the status of each device in the system. In multiuser system when more than one user requests for same I/O device occurs it manages the conflicting request by queuing up the I/O request. It allocates I/O devices and initiates I/O operation. On completion of I/O operation, the OS informs the process accordingly and reclaim the device. To provide buffering, caching, and spooling of data is the responsibility of device management. Interface to general device drivers and drivers to some specific hardware are provided by device management. 8.5.5

Secondary Storage Management

Main memory is too small to accommodate all data and programs, and because the data that it holds are lost when power is lost, the computer system must provide secondary storage to back up main memory.

Most

programs – including compilers assemblers, sort routines, editors, and formatters are stored on a disk until loaded into memory and use the disk as both the source and destination of their processing.

Operating system uses the concept of virtual memory to bring the data from secondary storage device to main memory. The proper management of disk storage is an important task. Operating system keeps all the information of secondary storage regarding free space, allocation mechanism and disk scheduling. Entire speed of operation of a computer may stuck if management of the secondary storage is inefficient. 8.5.6

Networking

A distributed system is

a collection of processors that do not share memory, peripheral devices, or a clock.

Each processor

in the distributed system

has its own local memory and clock,

and

the processors communicate with one another through various communication lines, such as high-speed buses

or

networks. Processors in

a distributed system may include small micro-processors, workstations, minicomputers, large,

and general-purpose computer systems.

Processors in the system are connected through a communication network, which can be configured in a number of ways.

A distributed system collects physically separate, possibly heterogeneous, systems into a single coherent

system, providing the user with access to the various resources that

the system maintains. 8.5.7 Protection System When

a computer system has multiple users and allows the concurrent execution of multiple processes, then the

various processes must be protected from one another's activities.

Protection is a

mechanism for controlling

the access of programs, processes or users to the resources defined by a computer system. 8.5.8

Command Interpreter System Command Interpreter System is the interface between user and the operating system. Many commands are given to the operating system by control statements. When a new job is started in a batch system, or when a user logs on to a time-shared system, a program that reads and interprets control statements is executed automatically. This program is called the control card interpreter or the command line interpreter.

Its function is to get the next command statement and execute it. Command statements deal with process creation and management, I/O handling, secondary storage management, main memory management, file system access, protection and networking etc. as per the requirement The key components of an OS are considered to be the built in device drivers which enable various hardware devices to work without the need to install the device drivers of the particular hardware and the basic features we need to get started, for example the windows 95 and 98 operating systems did not have even the drivers built in for setting up a graphic display which means that, until the graphic drivers are installed, the pictures on the monitor do not display enough colors. The inbuilt Components that are included in an OS, differs greatly with each operating system. For example, windows and Mac offer users a lot of in built programs, while operating systems like Linux do not offer the same number of packages built in their OS 8.6

Operating System Services

An operating system provides an environment for the execution of programs.

Common class of services provide by

the operating system are as follows: ➢

Program execution

➢ I/O operation ➢ File system manipulation ➢ Communications ➢ Error detection ➢ Resource allocation ➢ Accounting ➢ Protection 8.7

System Calls System calls provide the interface between a running program

and the operating system. These calls are generally available as assembly-language instructions.

Some languages defined to replace assembly language for systems programming allow system calls to be made directly (e.g., C, C++). Three general methods are used to pass parameters between a running program and the operating system. ➢ Pass parameters in registers. ➢ Store

the parameters in a table in memory, and the table address is passed as a parameter in a register. ➢

Push (store) the parameters onto the stack by the program and pop off the stack by operating system.

System calls can be classified into five categories: ➢ Process control

o Create process, terminate process

o Load, execute, end, abort o

Get process attributes, set process attributes o Wait for time o Wait event, signal event

o

Allocate and free memory ➢ File management o Create file, delete file o Open, close o Read, write,

reposition o

Get file attributes, set file attributes

➢ Device management o Request device, release device o Read, write,

reposition o

Get device attributes, set device attributes o Logically attach or detach devices ➢ Information maintenance o Get time or date, set time or date o Get system data, set system data o Get process, file, or device attributes o

Set process, file, or device attributes ➢ Communications o Create, delete communication connection

o Send, receive messages o Transfer status information o Attach or detach remote devices 8.8

System Programs System programs provide a convenient environment for program development and execution. These

can be divided into: ➢ File

management: These programs

create, delete, copy, rename, print, dump, list, and manipulate files and directories. ➢ Status information: Programs which provide the status of the system such as date, time, available memory, disk space, number of users

etc. ➢ File modification: Text editors are the example of system program which can create or modify the contents of the files stored on the secondary storage. ➢

Programming language support: Compilers, assemblers, interpreters for common programming languages are often provided to the user with the operating system. ➢ Program loading and execution:

After the

program is assembled or compiled, it must be loaded into memory to be executed. Absolute loaders, relocatable loaders, linkage editors, and overlay loaders

help in program loading and execution.

➢

Communications: These programs

provide the mechanism for creating virtual connections among processes, users and

different computer systems. They allow users to send messages,

browse web pages, send email messages, to login remotely, or to transfer files from one machine to another. ➢ Application programs: Most of the operating systems provide some useful applications or system utilities for common problems such as web browsers, word processor, text formatters, spreadsheets, database systems, compiler compilers, plotting and analysis package, games etc. 8.9 Kernel The kernel of an operating system is the part, responsible for many operations. When a computer boots up, it goes through some initialization functions, such as checking memory. Then it loads the kernel and switches control to it. The kernel then starts up all the processes needed to communicate with the user and the rest of the environment (e.g. the LAN). The kernel is always loaded into memory, and kernel functions always run, handling processes, memory, files and devices. The traditional structure of a kernel is a layered system, such as UNIX. In this, all layers are part of the kernel, and each layer can talk to only a few other layers. Application programs and utilities reside above the kernel. 8.10 Device Drivers

A device driver can be considered as a translator. Its input consists of high-level commands

and

output consists of low level, hardware specific instructions that are used by the hardware controller, which interface the I/O device to the rest of the system. The device driver writes specific bit patterns to special locations in the I/O controller's memory to tell the controller on

which device location to act and what actions to take.

Fig.8.11 Device drivers interface the

operating system with various hardware devices

Summary System software manages the operation of computer itself and can be divided into two categories essential and desirable. Application program are application specific and are developed to solve the problems for users. Then basic functions of OS are to act as an extended machine and to act as a resource manager.

Operating systems can be divided based on the computers they control and applications they support such as Main Frame, Server, Multiprocessor, Real time OS, Single user single task, Single user multi task,

Single user multi task, Multi user, Embedded OS, Smart card OS, Distributed OS etc. A process is a program in execution which needs certain resources. MMU keeps track of those parts of memory which are currently being used and decides which processes to load when memory space becomes available. File Management System keeps information about a file, its size, ownership, status, its use etc. In general OS supports Single directory, two level directory and tree structured directory for managing the files. OS keeps track of the status of each device in the system using device management. Secondary storage management manages the disk storage efficiently.

A distributed system is

a collection of processors that do not share memory, peripheral devices, or a clock.

Processes must be protected from one another's activities; this is known as protection. This is a mechanism for controlling the access of programs. Command Interpreter System is the interface between user and the operating system. System calls provide the interface between a running program and the operating

system. System programs provide a convenient environment for program development and execution.

Questions 1. Enumerate four basic functions of process management. 2. Compare Personal OS and Embedded OS. 3. Enumerate four basic functions of I/O management. 4. Enumerate O.S services. 5. Enumerate the two basic functions of an Operating System. 6. Give the classification of Operating System. 7. List two advantages of kernel design approach. 8. List various OS components. 9.

What are the three major activities of an operating system

in regards to secondary-storage management? 10. Write the five major activities of an operating system in regards to file management.

11. What is the purpose of system calls? 12. What is the purpose of system programs?

UNIT 9: Process Management 9.0 Objectives 9.1 Introduction 9.2 Process Management Functions 9.3 Process States 9.4 Process Control Block (PCB) 9.5 Context Switching of Process 9.6 Threads 9.6.1 Advantages of Multi-threading 9.6.2 Types of Thread 9.6.3 Threading Issues 9.7 Process Scheduling 9.7.1 Types of Scheduler 9.7.2 Scheduling Algorithm Criteria/Goal/Metrics 9.7.3 Preemptive and Non-preemptive Scheduling 9.7.4 Scheduling Algorithm 9.7.5 Operating System Scheduler Implementations 9.8 Interprocess Communication (IPC) and Process Synchronization 9.9 Deadlock 9.9.1 Neccessary Conditions 9.9.2 Methods of Handling Deadlock 9.0 Objective This chapter is to familiarizes the readers with ➢ the basics of processes and the threads ➢ process and thread state ➢ process scheduling algorithms ➢ deadlock conditions and handling them 9.1 Introduction In the early days computers were capable of running only one program at a time and had complete

control of the system resources. After completing this program another program is loaded and executed but development in the field of operating system allowed to load the multiple programmes in main memory and execute them simultaneously. This not only improves system resource utilization but also provide basis for multiprocessing and thus increased speed of execution. Allowing number of programs to reside in the main memory at the same time is defined as Multiprogramming and execute number of program at the same time (simultaneously) is known as multiprocessing. We define program as set of instruction and execution of these instruction is the key concept in any computer system. This lead to the notion of a process, which is defined as program in execution. Sometimes a process is also defined as a unit of work in modern time-sharing system. A number of terms have been used to define this unit of work given to CPU for execution such as jobs, tasks. For example a user may wants to run simultaneously a word processor, web browser, and e-mail package. There are many system level activities which execute transparently to complete the given task such as allocation of memory to a requesting process. Each of these activities is termed as process. The process is dynamic version of task performing action defined by program. The term job or process can be used interchangeably. It should be clear

that program itself is not a process. Program is a passive entity,

which is some times known as the text section whereas process is an active entity

and represented by its state. The state of process at any instant is generally defined by value in program counter, contents of the processor's register, program stack which contains temporary data and a data section which contains global variable, etc will be discussed in subsequent sections. Any process needs certain resources to complete its given task such as CPU time, memory, files, I/O devices and data structures. In general process management department of OS is responsible for the following activity in connection with process & thread management – Creation, Deletion, Scheduling, Synchronization, Inter Process Communication, Deadlock handling, etc. In the following section we have listed some important functions. 9.2 Process Management Functions ➢ Creating and removing process. ➢ Controlling the progress of processes, that is ensuring that each logically enabled process makes progress towards its completion at a positive rate. ➢ Handling exceptional conditions arising during the execution of a process, including interrupts and arithmetic errors. ➢ Allocation of resources among processes. ➢ Providing means of communicating messages (Inter Process Communication) or signals among

processes. ➢ Process synchronization. ➢ Process scheduling and managing swapping between main memory and secondary storage when not enough memory is available for executing processes. 9.3 Process States A process goes through a series of discrete states caused by various events. When a process executes, it changes state and move from one state to another

state. The state of the process depends on the current activity of that process. Each process must be in any one of following states: ➢ New: The

newly created process is

always in the

new state. In this state process reside in the memory and the execution is not yet started. ➢ Ready: When the process is ready to execute after completion of the waiting state or new state then process goes to the Ready state. In this state process is not actually executing, it is waiting for allocation CPU. ➢ Running: Process is being executing by CPU. ➢ Waiting: During the execution of the process if any event occurs that causes temporary suspension of the process, then process goes to the waiting state. After completion of the event process again moved to ready queue. ➢ Terminated: When the executing process completes its execution, it leaves the system (normal termination). Another possibility of termination is when executing process result in some serious error causing forced termination of the process in between to maintain the system stability and consistency. This is termed as abnormal termination. Fig. 9.1 Various states of a Process and State transition. The state shown in above diagram may vary from one system to another. However, only one process can be in running state at any time, while many processes may in waiting or ready state.

9.4 Process Control Block (PCB) Each process in the operating system is represented by a PCB also known as Task Control Block (TCB). PCB contains the information associated with the specific process. In other words PCB serves as the responsibility for any information that is necessary to define the state of a process and required by the system during context switching. The information contained in PCB is: ➢

Process state: Process state may be new, ready, running, waiting, terminated and so on. ➢ Program counter:

76%	**MATCHING BLOCK 139/158**	W
It indicates the address of the next instruction to be executed for this		

current process. ➢ CPU registers: The number and type of the CPU registers depends on the architecture of the computer system. It includes the accumulator, other registers, stack pointers etc. ➢ CPU scheduling information: This information includes the process priorities, pointer to scheduling queues and scheduling parameters. ➢ Memory-management information: This information can include value of the base register and limit registers, the page tables, or the segment tables depending on the memory management scheme used by the operating system. ➢ Accounting information: This information includes the amount of CPU and other resources usage time, time limits, account numbers, job or process number etc. ➢ I/O status information: This information includes the list of I/O devices allotted to the process, list of open files etc. Fig. 9.2 Process Control Block (PCB) 9.5 Context Switching of Process Switching the CPU from one process to another process is defined as context switching. This requires saving

the state of the old process and loading the saved state for the new process.

Therefore, context switching is a costly activity and must be minimized to reduce the time overheads i.e. wastage of CPU time as the system does no useful work while switching. The context of a process is represented by PCB of that process. Context switching times are highly dependent on machine hardware support such as memory speed, the number of CPU registers to be saved, etc. Typical speed ranges from 1 to 1000 microsecond. Till now we have discussed about the process model of computer system. In the following section we will introduce about the Threaded model. The key difference is the process model provides a single thread of control while threaded model of computer system gives multiple thread of control. Nearly, all modern operating systems support multithreading to improve performance and support multitasking. The key concepts which are applicable to process management are also applicable to thread management. 9.6 Threads Threads are generally defined as Light Weight Processes (LWP) and are a basics unit of the CPU utilization. It compromises a thread ID, a program counter, a register set, and a stack. It shares with other thread

belonging to the same process its code section, data section, and other operating system resources, such as open files and signals

therefore called Light Weight Process. A traditional process may be termed as Heavyweight and has a single thread of control. If a process contains of multiple thread of control, then it can perform more then one task at the same time (Multitasking). Now-a-day's most of the software packages are designed to support multithreading, whether they are system software or application software. Typically, an application is implemented as process with multiple threads of control. For example, consider a web browser may have one thread for display text or images and another thread for retrieve data from the network. If we consider word processer (MS-Office) then one thread may be waiting for characters from the keyboard typed by the user, second thread may be performing spelling and grammar checking, third thread may be launched to save the file, etc. following figure depict the key difference between traditional process model and modern multithreaded model of a computer system.

Fig.9.3 Single and multithreaded process 9.6.1 Advantages of Multithreading The multithreaded programming and system support provides the following major advantages: 1. Responsiveness: Multithreaded interactive applications improve the responsiveness as they may allow a program to continue running even if any part of the program is blocked or it's performing any other task. 2.

Resource sharing:

Threads shares the memory and other resources of the process to which they belong. The benefit of the code sharing that it allows the multiple threads to process in the same address space. 3. Economy: Creating and managing process is

costlier than threads in terms of time and memory (resources). This evident from the fact that threads allow sharing of common resources and address space of the process to which they belong. 4. Utilization of the multiprocessor architectures:

A single threaded process can only run on one CPU, no matter how many

CPUs (processors) are available at that time. But multithreading allow execution of different treads belonging to the same process to execute on different available CPU simultaneously i.e. in parallel. The degree of parallelization depends on the availability of number of free processors and number of threads currently available in the system memory. Multithreading in multiprocessor systems increases concurrency of execution hence faster results.

9.6.2 Types of Thread Generally, threads are classified as user threads and kernel thread. The threads runs in the user space are called user threads and the threads running in the kernel space are called kernel threads. 1. User Threads: Supported above the kernel and are implemented by the thread library at the user level. The library provides support for thread creation, scheduling, and management with no support from kernel. Because the kernel is unaware of the user level threads. All thread creation and scheduling is done in the user space without the need of kernel interaction. User threads are fast to create and easier to manage. The drawback of user level threads is that if the kernel is single threaded and any user thread performs a blocking system call then it will cause the entire process to block, even if other threads are available to run in the application. Examples of user thread libraries are POSIX

P threads,

Mach C-threads, Solaris 2UI thread. 2. Kernel Threads: Kernel threads are directly supported by the kernel. The thread creation, scheduling and management etc. are performed by the kernel and execute in kernel space. Because the thread management is carried out by kernel itself, the kernel threads are slower to create and manage than user threads. If a kernel

thread performs a blocking system call, the kernel can schedule another thread

unlike in the case of user threads where all threads are blocked. In this case, supporting multiprocessor architecture is easier because kernel can schedule threads on different processors. Examples of kernel threads are Windows 95/ 98/NT/2000, Solaris, Tru64 UNIX, BeOS, and Linux. 9.6.3 Threading Issues In this section some important issues related to thread are discussed. Thread Creation Generally same system calls used for process creation can be used for threads, but implementation may differ from one OS to another. For example, if one thread in a program calls fork system call, does the new process duplicate all threads or is the new process single threaded? Some UNIX systems have chosen to have two version of fork, one that duplicates all threads and another that duplicate the thread that invoked the fork system call.

Another system call

related to this is exec. Thread Cancellation Task of termination of thread before it has completed is called thread cancellation.

For example multiple threads are concurrently searching through a database and one thread returns the result the remaining

threads might be

cancelled.

A thread that is to be cancelled is often referred to as

target thread.

Cancellation of a target thread may occur in two different ways: • Asynchronous Cancellation: The target thread is immediately terminated. Most operating system allows a process or thread to be cancelled asynchronously. • Deferred cancellation: The target thread can periodically check if it should terminate, allowing target thread to terminate in a graceful manner. This method of cancellation allows the system to determine the safe state or point called cancellation point. Operating systems implementing Pthread API allow this mode of cancellation. Thread specific data Thread belonging to a process shares the data of a process. However each thread may need its own copy of certain data in some circumstances that is called thread specific data 9.7 Processes

Scheduling CPU scheduling is the basis of multiprogrammed operating systems.

Scheduling is defined as process of ordering the uses of system resources in an efficient and coordinated manner. Scheduling is a funda- mental function of operating system. Generally, processes in the system are divided into two categories: (a) I/O bound - spend most of their time waiting for I/O. (b) CPU bound or compute bound - spend most of their time in computing. Process execution starts with a CPU burst and then followed b I/O burst. The process alternated between CPU burst and I/O burst as per requirement until it finishes, this has been illustrated by following example:

>CPU burst starts< load store add store read >file< >CPU burst ends< [wait for I/O] >CPU burst starts< store incent index write >file< >CPU burst ends< [wait for I/O]

>CPU burst starts< load >store< add>store< read >store< >CPU burst ends< CPU Scheduler: − CPU scheduler is a software module which selects one of the ready processes for execution from ready queue according to the scheduling algorithm used and schedules it for execution. Dispatcher: - This is the part of the scheduler whose main function is to give

control of the CPU to the process selected by the short-term scheduler.

It should be very fast. The main functions of dispatcher are: • switching context of process or thread • switching to user mode or kernel

mode • jumping to proper location in user program to restart that program.

Dispatch Latency: -

The time taken by the dispatcher to stop one process and start running another process is known as dispatch latency.

It should be as small as possible. 9.7.1 Type of Scheduler Operating systems may feature up to three distinct types of schedulers: a long-term scheduler (also known as an admission scheduler), a mid-term or medium-

term scheduler and a short-term scheduler. (a) Long-term scheduler or job-scheduler

The process is

spooled to a mass storage device (hard disk) where they are kept for later execution.

It selects process

from this pool and loads them into memory for execution. The long-term, or admission, scheduler

decides which jobs or processes are to be admitted to the ready queue; that is, when an attempt is made to execute a program, its admission to the set of currently executing processes is either authorized or delayed by the long-term scheduler. Thus, this scheduler dictates what processes are to run on a system and the degree of concurrency to be supported at any one time - i.e. whether a high or low amount of processes are to be executed concurrently, and how the split between I/O intensive and CPU intensive processes is to be handled. Typically for a desktop computer, there is no long-term scheduler as such, and processes are admitted to the system automatically. However this type of scheduling is very important for a real time system, as the system's ability to meet process deadlines may be compromised by the slowdowns and contention resulting from the admission of more processes than the system can safely handle. It is generally invoked less frequently and controls degree of multiprogramming. (b) Medium Term Scheduler: The mid-term scheduler, present in all systems with virtual memory, temporarily removes processes from main memory and places them on secondary memory (such as a disk drive) or vice versa. This is commonly referred to as "swapping out" or "swapping in" (also incorrectly as "paging out" or "paging in"). In other words, removing the process from memory and some later time reintroducing the process into the memory to start the execution from the point where it left off, is called swapping. The mid-term scheduler may decide to swap out a process which has not been active for some time, or a process which has a low priority, or a process which is page faulting frequently, or a process which is taking up a large amount of memory in order to free up main memory for other processes, swapping the process back in later when more memory is available, or when the process has been unblocked and is no longer waiting for a resource or

swapping may be necessary to improve the process mix or

job mix. In many systems today (those that support mapping virtual address space to secondary storage other than the swap file), the mid-term scheduler may actually perform the role of the long-term scheduler, by treating binaries as "swapped out processes" upon their execution. In this way, when a segment of the binary is required, it can be swapped in on demand, or "lazy loaded". (c) The short - term scheduler or CPU scheduler The short-term scheduler (also known as the dispatcher) selects a job

from among the processes that are ready to execute and allocate CPU to one of them.

Thus the short-term scheduler

makes scheduling decisions much more frequently than the long-term or mid-term schedulers - a scheduling decision will at a minimum have to be made after every time slice, and these are very short. This scheduler can be preemptive, implying that it is capable of forcibly removing processes from a CPU when it decides to allocate that CPU to another process, or non-preemptive, in which case the scheduler is not allowed to snatch the CPU forcibly from the process, unless it releases the same. 9.7.2 Scheduling Algorithm Criteria/Goals/Metrics A large number of scheduling algorithm is present. The choice of good scheduling algorithm depends upon the application and the environment in which the system has to perform. These goals can be put in two categories one: - common to all and second, application or environment dependent such as (batch

processing, interactive, real-time systems). The various criteria includes: - CPU Utilization: - Should be as high as possible. The range is 0% to 100%. Throughput: - Number of processes complete per unit time is defined as throughput. Should be high or attempt are to be made to maximize this.

68%	**MATCHING BLOCK 140/158**	W

Turnaround Time: - The interval from the time of submission of a process to the time of completion is the

---

Turnaround time.

It measures how long the average user has to wait for the output. It should be small. Turnaround Time = waiting time in memory + waiting time in ready queue + executing in CPU + I/O time

Waiting time: - This

is the sum of all the periods spend waiting in the ready queue. Response time: -

For interactive systems, it is the measure of

time from the submission of a request until the first response is produced

i.e., amount of time

it takes to start responding, but not the time that it takes to output that response.

For other systems or in general, it can be defined as the difference between finish time and the arrival time of a process. Proportionality: - This criterion is related to user's expectations i.e. if an event takes place more time than a user expects, the user may feel irritated. So scheduling algorithms may be designed to satisfy user requirements. Such as- Minimize average value of any parameter such as waiting time; in

some cases we want

to optimize minimum or maximum values, rather than the average for

good interactive services; in interactive system some analyst suggests the minimization of variance in response time is more important. The common goals for any scheduling approach or algorithm are: • Fairness: - each process should get a fair share of the CPU. • Policy Enforcement: - system should ensure the stated policy is carried out. • Balance: - keeping usage of all system good i.e. good resource utilization • Timeliness: - meeting timing requirement (timing constraints such as deadlines) is important criteria in Real Time System.

9.7.3 Preemptive and Non-preemptive Scheduling 9.7.4

Scheduling

Algorithm There is large number of scheduling algorithms available in scheduling processes and threads. The selection of algorithm depends on the application for which it is to be used. These algorithms can be classified along many dimensions but only some common classes are outlined here. This depends whether algorithm is: • To be used for single processor or multiprocessor system. • Preemptive or non-preemptive; • For Real-time application or non-real-time applications; • Static priority based (does not change at run time, once assigned remain fixed) or dynamic priority based (priority can change at run time i.e. different instances of a process may have different priority) In computer science, a scheduling algorithm is the method by which threads or processes are given access to system resources, usually processor time. The need for a scheduling algorithm arises from the requirement for most modern system to perform multitasking, or execute more than one process to support pseudo parallelism (gaining illusion that multiple programs are executed at the time on a single CPU) or multiprocessing on recent multi-core architecture (having more than one processor CPU in the same processor). In this section we will discuss some common scheduling algorithms. For more algorithms you are advised to refer any operating system book given at the end of the chapter. The following scheduling algorithms are described:

Preemptive scheduling Non-Preemptive scheduling In this case scheduling algorithms picks a process and lets it run for a maximum of some time. if it is still running at the end of the time interval it will be suspended and scheduler picks another process to run. The scheduling algorithm picks a process to run and then just let it run until it blocks (earlier on I/O or waiting for another event/process) or until it voluntarily releases the CPU. CPU is taken off by force. Not be forcibly suspended. Result in overhead because needs context switching. Low scheduling overhead. Makes the kernel design more complicated. Simple design Enforcing the data consistency is complicated. Simple Need special such as timer. No special hardware is required.

•

First Come First Served (FCFS) • Shortest Job Next (SJF) • Round-Robin Scheduling (RR) • Shortest Remaining Job First (SRJF) • Priority scheduling First Come First Served (FCFS) In this approach the process or thread which arrive first in the system is scheduled first therefore the called First Come First Served. This is generally implemented by a first-in- first-out (FIFO) queue. When a process is entered into

ready queue, its PCB is linked onto the tail of the queue. When the CPU is free, it is allocated to the process or thread at the head of the queue. This is simple in implementation and understanding. A Gantt Chart is used to pictorially depict the behaviour of a scheduling algorithm. The following example shows the behaviour of the algorithm. Example: Consider the following three processes as given below. Find their average waiting time and turnaround time. Process Arrival Time Burst Time P1 0 24 P2 0 3 P3 0 3 P1 P2 P3 0 24 27 30 Gantt chart

Proces s Arrival Time Burst Time Waitin g Time Turnaroun d Time P1 0 24 0 24 P2 0 3 24 27

P3 0 3 27 30 Averag e 51/3=1 7 81/3=27

Calculation: Step 1: To find the Waiting Time Waiting Time = Finishing Time − Burst Time − Arrival Time Average Waiting Time = (?Waiting Time)/No. of Processes Step 2: To find the Turn around Time Turn around Time = Finishing Time − Arrival Time Average Turn around Time = (?Turn around Time)/No. of Processes Disadvantages ➢ The average waiting time is generally very large. ➢ If we do not have proper mix of CPU and I/O bound jobs, this algorithm may lead to starvation (a process waiting for long time to get the CPU). For example consider one large CPU bound process (with large execution time) and many I/O bound (small CPU time) jobs. Now assume CPU bound job is executing and hold CPU, mean time all I/O bound process finish their I/O and wait for CPU. While these processes are waiting for CPU, all I/O devices are idle. When this job complete its CPU burst and doing I/O all other job complete there CPU burst, leaving CPU idle. When CPU bound process move back to CPU. Again all I/O bound processes end up in ready queue waiting for big CPU bound job to finish. This leads to Convoy effect, all other process are waiting

for one big process to get off CPU. This result in lower CPU and device utilization. ➢

Not suitable for time sharing systems as FCFS is non-preemptive in nature. Shortest Job First (SJF) This is also known as Shortest Job Next (SJN)) is a non-preemptive algorithm for process scheduling. The algorithm selects the process having smallest execution time and dispatches for execution next. This needs to sort the ready queue according to there execution time having process with smallest CPU burst at the head of the queue. This algorithm assumes that execution burst is known in advance. Shortest job first is also simple in implementation but gives very low average waiting time (probably

minimum). It also maximizes throughput. However, it has the potential for process starvation for processes which require large CPU burst or execution time, if processes with short CPU burst are continually added. Shortest job next scheduling is rarely used outside of specialized environments because it requires accurate estimations of the runtime of all processes that are waiting to execute. Example: Consider the following three processes as given below. Find their average waiting time and turn around time. Proces s Arrival Time Burst Time Order of execution of the process

Waitin g

Time Turn around Time A 0 8 B 0 4 B 0 4 C 4 8 C 0 4 D 8 12 D 0 4 A 12 20

Averag e 24/4=6 44/4=11

Calculation: Step 1: To find the Waiting Time Waiting Time = Finishing Time − Burst Time − Arrival Time Average Waiting Time = (?Waiting Time)/No. of Processes Step 2: To find the Turn around Time Turn around Time = Finishing Time − Arrival Time Average Turn around Time = (?Turn around Time)/No. of Processes Shortest Remaining Job First (SRJF) SRJF also known shortest remaining time first is a preemptive version of shortest job first sched- uling. In this scheduling algorithm, the process with the smallest amount of remaining CPU burst is selected for execution. Since the currently executing process is the one with the shortest amount of time remaining by definition, and since that time should only reduce as execution progresses, processes will always run until they complete or a new process is added with smaller CPU burst requirement.

Shortest remaining time is advantageous because short processes are handled very quickly. The system also requires very little overhead since it only makes a decision when a process completes or a new process is added, and when a new process is added the algorithm only needs to compare the currently executing process with the new process, ignoring all other processes currently waiting to execute. However, it has the potential for process starvation for processes which will require large CPU burst or execution time, if processes with short CPU burst are continually added. Like shortest job next scheduling, shortest remaining time scheduling is rarely used outside of specialized environments because it requires accurate estimations of the runtime of all processes that are waiting to execute. Example: Consider the following three processes as given below. Find their average waiting time and turn around time. Proces s Arrival Time Burst Time P1 0 8 P2 1 4 P3 2 9 P4 3 5 P1 P2 P4 P1 P3 0 1 5 10 17 26 Gantt chart Process Arrival Time Burst Time Waiting Time Turnaround Time P1 0 8 9 17 P2 1 4 0 4 P3 2 9 15 24 P4 3 5 2 7 Average 26/4=6.5 52/4=13 Calculation: Step 1: To find the Waiting Time

Waiting Time = Finishing Time − Burst Time − Arrival Time Average Waiting Time = (?Waiting Time)/No. of Processes Step 2: To find the Turn around Time Turn around Time = Finishing Time − Arrival Time Average Turn around Time = (? Turn around Time)/No. of Processes Round-Robin Scheduling Round-robin is one of the simplest scheduling algorithms, which assigns time slices to each process in equal portions and in order, handling all processes without priority. The name of the algorithm comes from the round-robin principle known from other fields, where each person takes an equal share of something in turn. This commonly used in time-sharing systems. Some important points about RR scheduling are outlined below: ➢

A small unit of time, called a time quantum or time slice is defined. Typical rage is (10- 100ms) ➢ Ready queue is treated as circular queue. ➢

Process at the head of a queue is dispatched every time. ➢ Process is preempted after the expiry of its current time slice and put at the tail of ready queue. Round-robin scheduling is both simple and easy to implement, and starvation-free. Round-robin scheduling can also be applied to other scheduling problems, such as data packet scheduling in computer networks. One important issue here is the selection of time slice, if time slice is very large the algorithm will behave similar to FCFS. On the other hand, if it is very small-RR approach is called processor sharing it appear that each user has its own processor with speed 1/n; where n is the number of processes. But this will result in large context switching overhead and turn around time may increase with small value time slice. Example: Consider the following three processes as given below. Find their average waiting time and turn around time Process Arrival Time Burst Time P1 0 8 P2 1 4 P3 2 9 P4 3 10 P5 0 15 Solution: Let us assume the time slice is 4.

P1 P5 P2 P3 P4 P1 P5 P3 P4 P5 P3 P4 P5 0 4 8 12 16 20 24 28 32 36 40 41 43 46

Gantt chart The calculation of average waiting time and turn around time is left as an exercise to students. Priority Scheduling The algorithms discussed above except Round-Robin are special case of the general priority scheduling. Where, priority of each process is inherent and based on one of the task parameter such as arrival time, exaction time. However, in priority scheduling algorithms the priority of each process is explicitly assigned based on some criteria such as importance of task or criticalness of tasks. The important features are outlined below: ➢ Here the priority each process is some fixed number. For example some fixed range of number may be (0-7) or (0-4095) are used as priorities ➢ System schedule higher priority process first for execution and if lower priority process is executing, scheduler preempts the same. ➢ Priorities can be defined internally or externally. Internally assigned priorities are based on some measurable quantity such as memory requirement, time limit,

number of open files and ratio of average I/O burst to average CPU burst.

While externally assigned priorities are external to the OS and based on factors such as importance of work, political, financial, etc. ➢ This paradigm of scheduling can be Preemptive or non-preemptive ➢ Priorities can be assigned statically or dynamically. Disadvantages: This algorithm may also result in starvation because lower priority process may not get the chance if higher priority processes are always present in the system. Solution to this problem is Aging. Aging is a technique of gradually (periodically) increasing the priority of processes that are waiting in the system for a long time, for example increment priority of victim process by one after each 30 minutes.

9.7.5 Operating System scheduler implementations Different computer operating systems implement different scheduling schemes. Very early MSDOS and Microsoft Windows systems were non-multitasking, and as such did not feature a scheduler. Windows 3.1-based operating systems use a simple non-preemptive scheduler which requires programmers to instruct their processes to "yield" (give up the CPU) in order for other processes to gain some CPU time. This provided primitive support for multitasking but did not provide more advanced scheduling options. Windows NT 4.0-based operating systems use a simple round robin, preemptive scheduler with priorities. Priorities in Windows NT 4.0 based systems range from 1 through to 31, with priorities 1 through 15 being "normal" priorities and priorities 16 through 31 being "real time" priorities, requiring administrator privileges to assign (although "real time" is a misnomer given that no Windows scheduler has support for real-time scheduling, as is the case with most desktop operating systems). Users can select 5 of these priorities to assign to a running application from the system manager, accessed through the alt control delete key combination. When choosing a process to dispatch (allocate a CPU to) the Windows NT 4.0 scheduler will simply select the highest priority process that is ready to run as the next process to be given CPU time, in a round robin fashion (so if there are three processes of the same priority, A, B and C, and a process of lower priority, process D, process A will be allocated the CPU, followed by process B, followed by process C - process D will be allocated the CPU if the other processes are blocking on a resource or are not waiting for the CPU for some other reason). In order to avoid starving low priority threads of CPU time, the Windows NT 4.0 scheduler will give a process that has not been executed in over 3 seconds a temporary priority "boost", elevating it to the top of the queue of processes to run and giving it a longer time slice (quantum) than is normal, ensuring that it is allocated CPU time for the next (longer) time slice. In Linux two separate scheduling algorithms are supported one for time sharing systems to provide fair preemptive scheduling and one for priority driven to support real-time task scheduling. Time sharing processes use prioritized, credit based algorithm. Real-time task scheduling uses FCFS and Round- Robin approach. For more details students are advised to refer any operating system book given at the end of the chapter. 9.8 Interprocess Communication (IPC) And Process Synchronization We have number of processes executing in the system simultaneously and these processes need to communicate to cooperate in execution and successful completion of the given task. IPC provide mechanism to allow processes to communicate and to synchronize their actions without sharing their address space.

The commonly used method for implementation of IPC is by message passing. The messages send by a process to another process can be of fixed or variable size. The implementation of fixed size message is simple and straight forward. Generally IPC facility provides two operations: ➢

Send (P, message) – send message to process P. ➢ Receive (message) – receive message from process Q.

There are number of issues in implementation of IPC such as direct or indirect communication, symmetric or asymmetric communication etc. which are not discussed here the interested student may refer the book on operating system.

A cooperating process is one that can affect or affected by other processes executing in the system. These processes need concurrent access to resources, which may lead to data inconsistency if not properly coordinated. This needs synchronization of processes of resource allocation so that system consistency is maintained. We will introduce some basic concept related to this problem. Another problem with process synchronization is race condition if access to non-sharable resources is not properly constrained. The race condition is defined, a situation where two or more processes are allowed to access and manipulate (read and write) some shared data (resource) concurrently and the final result depends who runs precisely when. The race condition may lead to data inconsistency. This can be explained by following example:

To guard against race condition, we need to ensure that only one process is allowed to access and manipulate the shared data. This needs process synchronization and coordination. Critical Section (CS) The part of a program where shared memory or any resource is accessed and manipulated (changing variable, updating a table or writing to a file) is called critical section. The important requirement is that when one process is executing in its critical section, no other process is allowed to enter in its critical section. Thus, the execution of critical section by processes must be mutually exclusive. This needs to design protocol/approaches to implement mutual exclusion or process synchronization. In general, each process wishing to enter in the critical section must request for permission and if granted enters into its CS and execute. After completing CS it gracefully exits the CS. Following fig. pictorially depict the structure of a process with CS. Do { ENTRY SECTION; // provide controlled access to CS CRITICAL SECTION; // process accessing shared resources EXIT SECTION; // process exit CS after performing necessary action

REMAINDER SECTION; } while (1); Fig. 9.4 General Structure of a typical process with critical section. We need four conditions to have good solution for processes to cooperate correctly and efficiently: ➢ Mutual Exclusion: no two processes are allowed to execute simultaneously in their critical section. ➢ No assumption should be made about the speed and number of CPUs. ➢ No process running outside the CS may block any other process. ➢ No process should have to wait forever to enter in its CS (bounded wait to avoid starvation). There are number of synchronization primitives used for process synchronization or enforcing mutual exclusion such as disabling interrupts, using hard or soft locks, TSL (Test and Set Lock) instruction, semaphores and monitors. Semaphore approach is commonly used in many operating systems to grant the controlled access to critical section. Semaphores are positive integer variables with some defined atomic operations (UP and DOWN). DOWN is an atomic operation which read the value of semaphore, tests it and if its value is positive non- zero decrement it, otherwise suspend (asked to sleep) the process invoking the DOWN operation. Later when some process does UP on this semaphore the suspended process is signaled (to awaken) and can be allowed to execute its critical section. The UP operation is also atomic in nature, it increments the semaphore by one and if one or more processes are sleeping on this one of them is allowed to complete its DOWN. 9.9 Deadlock A deadlock is a situation where in two or more competing actions are waiting for the other to finish, and thus neither ever does. It is often seen in a paradox like 'the chicken or the egg'. In the computer systems deadlock refers to a specific condition when every process in a set of processes

is waiting for an event that can be caused only by another process in the set.

We have studied that every process needs resources to complete the given task and number of resources in the system are finite. Then it is possible that if each process needs multiple resources and if the resource allocation is not coordinated properly, following situation may arise. Consider two processes A and B both needs printer and file in write mode, process A is allocated printer by OS and process B is allocated the file. Now process A will wait for the file and process B for printer neither of two can continue to execute, both are

blocked for indianite time leading to deadlock situation. This situation is defined as circular wait (see Necessary conditions), where each process in the circular chain is waiting for a resource held by another process in the chain. Deadlock is a common problem in multiprocessing or multithreading system, where many processes share specific type of mutually exclusive resources. Computers intended for the time-sharing and/or real-time systems are often equipped with a lock (soft or hard), which guarantees exclusive access to processes. Deadlocks are particularly troublesome because there is no general solution to avoid deadlocks. To further understand the problem of deadlock consider a situation where two people who are drawing diagrams needs both resources pencil and ruler at the same time, there is only one pencil and one ruler between them. If one person takes the pencil and the other takes the ruler, a deadlock will occurs. Because when the person holding the pencil needs the ruler, its request can not be granted because system does not have another ruler (the one is available is held by another person and cannot be released). Both requests can't be satisfied, so a deadlock occurs. 9.9.1 Necessary conditions There are four necessary conditions for a deadlock to occur, known as the Coffman conditions from their first description in a 1971 article by E. G. Coffman. These conditions must hold simultaneously for system to be in deadlock. 1.

Mutual exclusion condition: At least one resource must be held in exclusive mode (non-sharing mode). 2.

Hold and wait condition: Processes are allowed to hold any resources and while holding allowed to request for new resources. 3.

No preemption

condition: Resource cannot be preempted; a process holding a resource may release it voluntarily or after completion, but system cannot force for release of the same. 4. Circular wait condition: two or more processes form a circular chain where each process is waiting for a resource that is hold by the next process in the chain. 9.9.2 Method of Handling Deadlocks There are following possible ways to deal with problem of deadlocks in operating systems: ➢ Deadlock Prevention ➢ Deadlock Avoidance ➢ Allow the system to enter in the deadlock state, detect it and recover. Prevention Deadlocks can be prevented by ensuring that at least one of the above necessary four conditions is denied: • Denial of mutual exclusion condition This means that no process may have exclusive access to a resource. This condition cannot be satisfied as there are many resources which can be access only in exclusive mode such as printer. Sometimes mutual exclusion is also necessary to implement resource/process synchronization to maintain system consistency. • Denial of hold and wait condition If this condition is enforced, deadlock will never occur in the system. Here whenever a process request for a resource, it should not hold any other resource. To implement this possible ways are - each process should be allocated all its required resources before its execution starts; or allow a process to request required resources only when it holds no resource. • Denial of no preemption To ensure that no preemption condition should not hold, we can make use of

the following protocol: "If a process is holding some resources and

request for another resource that cannot be immediately allocated to it, then all resources currently being held are being preempted from that process". • Denial of circular wait condition To ensure that this condition should not hold, we can make use of the following protocol: "

Impose a total ordering of all resource types, and to require that each process requests resources in an increasing order of enumeration".

Circular wait condition is eliminated by assigning unique priority number with each resource. Processes may only request a resource if its priority is higher than all currently held resources. If the priority numbers are chosen wisely, this system may accommodate the need of many processes. However, it will not work in all cases. Avoidance Deadlock can be avoided if certain information about resource allocation is available in advance. For every new request, the system determines whether after granting the request system will enter into a safe or unsafe state (that could result in deadlock). The system grants request only when it leads to the

safe state. A state is safe if the system can allocate resources to each process in some order and still avoid a deadlock. In order to determine, next state will be safe or unsafe, it must know in advance number and type of all resources available, and process need. One known algorithm that is used for deadlock avoidance is the Banker's algorithm. In practical situations deadlock avoidance is not used because it is difficult to know resource requirement in advance. It is important to note that a process may be in unsafe state but may not result in a deadlock. The safe or unsafe state only refers to the ability of the system to enter in a deadlock state or not. Detection Often neither deadlock avoidance nor deadlock prevention may be used. Instead deadlock detection and recovery approach are used to deal with deadlock situation. In this case system permits the deadlock to occur in the system and then detect it. In the deadlock situation system must provide: • An algorithm that examines whether the deadlock has occurred. • An algorithm to recover from deadlock. •

When

should we invoke detection algorithm? How often a deadlock is likely to occur? – Invoking the deadlock detection algorithm for every request may be very costly. An alternative approach is to invoke the algorithm at less frequently or whenever CPU utilization drops below 40%. • How many processes will be affected by the deadlock when it happens? Recovery from deadlock To eliminate deadlocks, we may use following methods– • Abort all deadlocked processes: In this case OS has to discard all the partial executed processes and recomputed at

later stage. •

Abort one process at a time until the deadlock cycle is eliminated: In this

case pro- cesses are killed one by one until deadlock is removed. Overhead in this approach are more because deadlock detection algorithm has to be invoked again and again. To eliminate deadlocks using resource preemption we require: • Selecting a victim: selecting a process to be terminated. • Rollback: determine how much rollback is necessary to break deadlock. • Starvation: to ensure same process should not be selected again and again as victim. In general it is also possible to ignore the problem of deadlock completely, assuming deadlock never

occur in the system. It is stated that this solution is used by most operating systems like UNIX. Summary ➢ Process management is the part of the operating system to manage and coordinate the execution of various processes. These include creation, removal, process scheduling, process communication and synchronization, deadlock handling and many more. ➢ A process control block provides an abstracted view of a process. A process can be in any one of the state that is new, ready, running, waiting or terminated. ➢ Multithreading is an important feature supported by all modern operating systems. Threads are lightweight processes share code and data with their parent process. We have two types of threads user level and kernel level. ➢ Scheduling is an important function of process management which defines the order of execution of processes present in the system. The commonly used algorithms are FCFS, SJF, SRJF, Round Robin and priority scheduling. ➢ IPC provide

mechanism to allow processes to communicate and to synchronize their actions without sharing their address space. The commonly used method for implementation of IPC is by message passing. ➢

A set of processes are in deadlock state when every process in the set is waiting for an event that can be caused only by another process in the set.

The

deadlocks are handled by prevention, avoidance or detection followed by recovery mechanism. Questions 1. Enumerate basic functions of process management. 2. Draw state diagram of a process and label various transitions. 3. What is a Process Control Block? 4. Explain the need of process control block? 5. Give priority rule & one example of each of the following scheduling algorithm: (a) First Come First Serve (b) Shortest Job First (c) Shortest Remaining Job First (d) Priority Scheduling (e) Round Robin scheduling.

6. Define starvation and aging with respect to scheduling and its use? 7. Write short notes on: (a) Dispatcher (b) Dispatch Latency (c) Scheduling (d) Type of schedulers (e) Context Switching. (f) Process 8. What are necessary conditions for deadlock to occur in the system? 9. Denying hold & wait condition to prevent deadlock may suffer with starvation, Explain how? 10. How would you prevent deadlocks in the system? 11. Why deadlock avoidance is difficult to implement in practical operating systems? 12. Differentiate between user threads & kernel threads. 13. Enumerate advantages/benefits of multithreading? 14. Which scheduling algorithm is used in LINUX? 15. Explain why Asynchronous cancellation of thread may lead to problems. 16. Consider the set of the processes given in table-1 and find the average waiting time and turnaround time for the following scheduling algorithms (Give results in tabular format): (a) Shortest Job First (SJF) (b) Shortest Remaining Job First (SRJF) (c) FCFS (d) Round Robin scheduling with time slice (t=2 and t=4) Process id Arrival time Execution Time A 0 7 B 1 15 C 2 3 D 6 8 E 12 10 17. Discuss whether each of the following states is safe or unsafe. If a state is safe, show how it is possible for all processes to complete. If a state is unsafe, show how it is possible for deadlock to occur. (l (i) denotes allocation and m(i) denotes maximum demand). State A State B User i l(i) m(i) User i l(i) m(i) 1 3 6 1 5 8 2 4 7 2 3 9 3 5 8 3 4 8

4 0 3 Available = 3 Available = 2 18. Define Interprocess Communication (IPC) and its need. 19. What is critical section? Why process entry to critical section is restricted? 20. What are semaphores? Give there use.

UNIT 10: Memory Management 10.0 Objectives 10.1 Introduction 10.2 Address Binding 10.2.1 Logical vs Physical Address Space 10.2.2 Address Translation and Relocation 10.3 Memory Allocation 10.4 Fragmentation 10.4.1 Memory Compaction 10.5 Virtual Memory 10.5.1 Paging 10.5.2 Segmentation 10.5.3 Segmentation vs Paging 10.6 Demand Paging 10.7 Working Set Model 10.8 Prepaging 10.9 Thrashing 10.10 Page Replacement Algorithm 10.0 Objective After reading this unit, readers will be familiar with ➢ various memory management techniques ➢ concepts of logical and physical address space ➢ address binding, address mapping and memory allocation ➢ virtual memory management ➢ paging, segmentation, and segmentation with paging ➢ associative memory and working set model 10.1 Introduction Memory is an important resource that must be carefully managed. While the average home computer now a days has a thousand times as much memory as the IBM 7094, the largest computer in the world

in the early 1960's, programs are getting bigger, faster then memories. Ideally, what every programmer would like is an infinitely large and fast memory that is also non volatile (Does not loose its contents when the electric power fails). Most computers have a memory hierarchy, with a small amount of very fast, expensive, volatile cache memory, tens of megabytes of medium-speed, medium-price, volatile main memory (RAM), and tens or hundreds of gigabytes of slow, cheap, non- volatile disk storage. It is the job of the operating system to coordinate how these memories are used. Memory management is not just a software task. The operating system requires hardware support to implement any memory management scheme. This hardware is typically protected from user access. All modern computer systems have memory management unit (MMU) to manage all functions related to memory management. The part of the operating system that manages the memory is called the memory manager. Some of the important functions of the memory management unit are summarizes below: ➢

76%   **MATCHING BLOCK 141/158**   W

Keep track of which parts of memory are in use. ➢ Allocate memory to processes when they need it. ➢ Deallocate

---

memory from processes when they have completed. ➢ Manage swapping between main memory and disk when memory is not enough to hold all the processes. ➢ Protecting memory allocated to each process from undesired access by other processes or user. While the simple memory management schemes such as Memory Management without Swapping or Paging are no longer used on desktop computers, they are still used in some palmtop, embedded, and smart card systems. For this reason, they are still worth studying the two class of Memory Management System- (a) Those that move processes back and forth between main memory and disk during execute and (b) Those that do not. The simplest possible memory management scheme is to have just one process in memory at a time and allow only that process to use all the memory. This system has its limitation. It provides no services to user. The user has complete control over the computer, but the OS has no control over interrupts, no resident monitor to process system call and errors, etc. It is generally used only on dedicated systems, where the users require flexibility and simplicity and willing to program their own support

routine. No dedicated space is provided for OS. The usual technique used on simple microcomputer is to divide the memory in between the OS and a simple user program. The OS may be at the bottom of memory in RAM. Fig.10.1 Operating System in Main Memory The more advance schemes used in memory management will be discussed in the following sections. Now we introduce some important concept required to understand these memory management schemes. 10.2 Address Binding Addresses in the source program are generally in symbolic form. Compiler binds these symbolic addresses to relocatable addresses and finally loader binds these

addresses to absolute addresses. Therefore binding is a mapping from one address space to another (such as logical address to physical address will be discussed in the next section).

Binding of instructions and data to memory addresses can be done at: •

Compile time:

In this form of address binding the compiler knows about the location of the process to be loaded into the memory. Therefore compiler generates absolute addresses. • Load Time: If at compile time the actual location in the physical memory is not known then the compiler must generate relocatable code and binding is delayed till load time. The advantage of this scheme is if starting address of the process changes then we have to reload the base register with new starting address to bind these relocatable addresses to absolute addresses. • Execution Time: Here the binding of relocatable addresses to absolute addresses (Physical Addresses) is done at the execution time of the programme. This needs special hardware support. Most general purpose OS use this method. 10.2.1 Logical versus Physical address space Physical memory in a computer system is a hardware device. The physical memory is divided up into small units called memory cells. Each memory cell is given a unique name/number called physical address. These physical addresses together form the physical address space. For example, if a computer system has 512 MB (2 29 Bytes) of memory, then the physical address space will run from [0 to (2 29 -1)] ➢ An address generated by the CPU (user programme) is referred as logical address and

an address seen by the memory unit and loaded into memory address register is known as physical address. ➢ Logical Address also known as Virtual Address. ➢ Set of all physical address are known as Physical Address Space and set of all logical address generated by a program is Logical Address Space. ➢

Computer system has only one physical address space but can have several logical address spaces. ➢ Mapping of logical address onto physical address is done by hardware unit i.e. Memory Management Unit. 10.2.2 Address Translation and Relocation In the following diagram translation of logical address to Physical Address is depicted. It also illustrated the process of Relocation of relocatable code generated at compile time as explained above. To perform relocation, linker must include in the binary program, a list or bit map telling which instructions are to be relocated. To facilitated relocation the memory management unit has relocation or base register to convert logical address into physical address (PA). For example if the value in the base register is 2000 and CPU generates logical address (LA) of given instruction 400 then the Physical address will be 2400(PA = Base Register value + LA). Intel 80x86 families uses 4 relocation registers when loading and running processes. As shown in the diagram the limit register provides the protection against the generation of logical address outside the range of logical addresses (illegal addresses). The relocation register contains the smallest physical address of a user process. While Limit register provides the largest value of physical address which can be use by a user process. For example if relocatable register contains value 5000 and limit register contains 600 then user process address space will be from 5000 to 5600.

10.2.3 Dynamic Loading In Dynamic loading a routine is loaded at the time of calling (Execution). This provides better memory space utilization as in this method an unused routine never loaded into the memory.

This method is useful when large amounts of code are needed to handle infrequently occurring cases, like error routines. Dynamic loading does not require speciall support from the operating system.

It is the responsibility of the user to design their programs to take advantage of such a method. Operating system may help the programmer, however, by providing library routines to implement dynamic loading. 10.3

Memory Allocation Allocation of memory space is required both for operating system and user processes in most efficient manner. Therefore the MMU of OS must provide mechanism for memory allocation to the requesting processes. One of the simplest methods for memory allocation

is to divide memory into several fixed sized partitions. Each partition may contain one process,

however, in the modern OS this scheme is too inflexible and inefficient because this may lead to wastage of memory in fragmentation and do not cater to the need of accommodating variable size processes or data. Therefore the main memory is considered as one large block or hole. When a process arrives and request for memory space the MMU search for hole large enough to accommodate this process. If it finds one, MMU allocate only as much memory as needed by this process and keep the rest for future request (dividing this large hole into two pieces if larger then the process size). There is large number of algorithms used for memory allocation. Some of the common algorithms used for this purpose are First Fit, Best Fit and Worst Fit. Some of the common algorithms used for this purpose are First Fit, Best Fit and Worst Fit. ➢ First Fit: Allocate the first hole big enough to accommodate the requesting process. ➢ Best Fit: The entire list is searched and smallest hole that is big enough to accommodate the requesting process is allocated. ➢ Worst Fit: The entire list is searched and largest hole that is big enough to accommodate the requesting process is allocated. Example 1: Given memory partitions/holes of 100KB, 500KB, 200KB, 30KB, and 600KB. Processes request for allocation of memory space of 212KB, 417KB, 112KB, and 280KB in order. Show how the memory will be allocated to different process using first fit, best fit and worst fit algorithms?

Solution: First Fit: Process (in KB) Allocated to memory space (in KB) 212 500 417 600 112 500-212 280 Not allocated Best Fit: Process (in KB) Allocated to memory space (in KB) 212 500 417 600 112 600-417 280 500-212 Worst Fit: Process (in KB) Allocated to memory space (in KB) 212 600 417 500 112 600-212 280 Not allocated 10.4 Fragmentation The above memory allocation algorithms suffer from the problem of fragmentation. The memory fragmentation is defined as conversion of memory into unusable small holes. This is of two types: 1. External Fragmentation – External Fragmentation exist when enough

total memory space exists to satisfy a request, but it is not contiguous.

Memory is fragmented into number of small holes. 2. Internal Fragmentation – Generally occurs when memory is divided into fixed sized units such as pages and processes are allocated memory spaces in the form of number of such units or

pages. With this scheme generally the last page of a process is not fully occupied and leaves unused space. This unused space is termed as internal fragmentation. 10.4.1 Memory Compaction When memory is divided into multiple unusable small holes due to external fragmentation, it is possible to combine them all into one big hole by moving all the processes downward as far as possible (

Shuffle

memory contents to place all free memory together in one large block).

This technique is known as memory compaction.

Compaction

is possible only if relocation is dynamic, and is done at execution time. 10.5

Virtual Memory Many years ago people were first confronted with programs that were too big to fit in the available memory. The memory was small and expensive. For example: IBM 650 (late 1950) have memory of only 2000 words and PDP-1 (TSS) have memory of 4096 (4k) *18 bit words. Traditional solution is to use secondary memory and divide the programme into a number of pieces that can fit into memory called overlays. Overlay would start running first. When it was done, it would call another overlay. Some overlay systems were highly complex, allowing multiple overlays in memory at once. The overlays were kept on the disk and swapped in and out of memory. The Programmer- • Was responsive for breaking the programme into overlays. • Deciding where in the secondary memory each overlay was to be kept. • Arrange for the transport of overlays between main memory and secondary memory. • Manage whole overlays process without any help from the computer system. Although the actual work of swapping (movement between main memory and secondary storage) overlays in and out was done by the OS later, but the work of splitting the program into pieces had to be done by the programmer. Splitting up large programs into small, modular pieces was time consuming and boring. Therefore the concept of virtual memory was introduced to transfers this job to computer system. The basic idea behind virtual memory is that the combined size of the program, data, and stack may exceed the amount of physical memory available for it. The operating system keeps those parts of the program currently in use, in main memory, and the rest is onto the disk. For example, a 16-MB program can run

on a machine with 4-MB main memory by carefully choosing which 4 MB to keep in memory at each instant, with pieces of program being swapped between disk and memory as needed. Thus, Virtual memory management is a technique used in computer systems for performing automated overlay process without the programmer even knowing that it was happening. The idea put forth by Manchester group (England 1961, Fotheringham) was to separate the concept of Address Space and Memory Locations. The following example will help in understanding the underlying principle of VM. Address Space: - The number of addressable words which the computer (CPU) can address, depends on the number of address bit in its address. A computer system having 16-bit address has address space 0 to (2 16 -1) or 0 to 65535. If this computer have only 4K physical memory, then useful address space will be (0 – 4095) and address space from 4096 to 65535 can not be used. If program jumps to an address from 8192 to 12287 on a system without VM, the program would cause an error and print the message Non-existence memory. But on system with VM following sequence of steps would occur: ➢ The content of memory would be saved in the secondary memory. ➢ Word 8192 to 12287 would be located in the secondary memory. ➢ Word 8192 to 12287 would be loaded in the main memory by mapping onto physical address space (0 to 4095). ➢ The address map would be changed to map addresses 8192 to 12287 onto memory location 0 to 4095. ➢ Execution would continue as though nothing unusual had happened. Logical or Virtual address space: - The range of addresses which the program can use or refer i.e. 0 to 65535. Actual/Physical address space: - It is the range of addresses hardwired to the main memory or the range of addresses actually available in the main memory i.e. (0 – 4095). Memory map: - Relates virtual address to the physical address.

Fig. 10.3 Mapping the logical address of user program onto physical address Thus VM

is the separation of user logical memory

from physical memory. This separation allows an extremely large virtual memory

is provided for programmers when physical memory is smaller in size. This makes the task of programmers very easy as discussed above. In addition to this VM also allows sharing of files and memory between several processes and makes implementation of memory protection much easier. Virtual memory can be implemented by one of the following techniques: 1. Paging: 2. Segmentation 3. Segmentation with Paging 10.5.1 Paging Paging is a memory management scheme in which memories as well as processes are divided into fixed size of blocks. Paging is the schema that permits the logical address space of a process to be noncontiguous. Physical memory broken

---

50% **MATCHING BLOCK 142/158** W

into fixed sized blocks called Page frames. Logical memory address space/process is also broken into blocks of same size

---

as page frame called Pages.
When a process is to be executed, its pages are loaded into
any available memory frames from
secondary storage. The secondary storage is also
divided into fixed size blocks that are
of the same size as the memory frames. The
page size is defined by the hardware and typically a power of two. The size of page typically lies between 512 bytes and 16 MB, depending upon computer architecture.
d(page offset) p(page number)
Fig. 10.4 Address Translation Architecture Page Table The operating system stores the address translation tables for mapping virtual address to physical address in a data structure known as a
page table. ➢
Page table is kept in main memory. ➢ Page-table base register (PTBR) points to the page table. ➢ Page-table length register (PTLR) indicates size of the page table. ➢ In this scheme every data/instruction access requires two memory accesses. One for the page table and one for the data/instruction. ➢ The two memory access problem can be solved by the use of a special fast-lookup hardware cache called associative
memory ➢
Address generated by
CPU is divided into: Page number (p) – used as an index into a page table which contains base address of each page in physical memory.
Page offset (d) – combined with base address

100% **MATCHING BLOCK 143/158** W

to define the physical memory address that is sent to the memory unit.

If the size
f the
logical address space is 2 m , and a page size is 2 n
addressing units (bytes or words),
then the high order m-n bits of a logical address designate the page number, and the n low order bits designate the page offset.
m-n n Example: Show how the logical memory can be mapped in physical memory for the following page size and primary memory. Page size = 2bytes Physical memory = 16bytes
Fig. 10.5
Paging example for a 16 byte memory with a 2 byte page Solution: Logical Address =Frame Number X page size + Offset Logical address 0 maps to the physical address 8 = (4x2+0) Logical address 1 maps to the physical address 9 = (4x2+1) Logical address 5 maps to the physical address 1 = (0x2+1) Associative memory (Hardware support) In the simplest case page table is implemented with the help of dedicated registers.

100% **MATCHING BLOCK 144/158** SA Operating System-edit.pdf (D143421170)

Page table is kept in main memory and a page table base register (PTBR) points to the page table.

The problem is there

with this approach is much time require to access a user memory location. If we want to access some location i, we must get index into the page table from the PTBR offset. This task requires a memory access.

Now we get the page number from the page table and combine

with the page offset to produce the actual address. With this scheme two memory accesses are needed to access a byte.

The most common

solution to this problem is to use a special small fast lookup hardware cache called Translation Look-aside Buffer (TLB). The TLB is associative high speed memory. Each entry in the TLB consists of two parts a key and a value. If the

item is found the corresponding value field is returned. TLB search is fast but hardware is expensive. Generally the number f entries in TLB are small.

TLB contains only few of the page table entries. When a logical address is generated by the CPU, its page number is given to the TLB. If the page number is found its frame number is immediately available and used to access required memory. If the page number is not in the TLB (TLB miss) a memory reference to the page table is used to get the frame number

the page table is used to get the frame number from the page

table. This frame can be used to access the main memory. This page number and frame number is now added to the TLB for the future reference. If the TLB is full then OS must replace one of the entry from the TLB. Some entries from the TLB cannot be removed

are wired down. Some TLBs stores address space identifiers (ASIDs) in each entry

of the

TLB. An ASID uniquely identifies each process and is used to provide address space protection for that process.

Fig. 10.6 Paging with Translation Look-aside Buffer HIT Ratio:

Let Hit Ratio=
h =< Miss Ratio=(1-h) TLB search takes ts seconds Memory access time tm seconds If page is in TLB (Hit) Memory access time = (ts + tm)

If page is not in TLB (miss) Memory access time=(ts + 2tm) (tm to read page from memory ts to read data) Effective access time= (ts + tm)h + (ts + 2tm)(1-h) Example: Find the effective access time if the Hit ratio, TLB search and memory access time is given as: Hit Ratio 90% t s =15ns t m =80ns Solution: Effective access time=.90X(15+80) +.10X(15+2X80) =.9X95 +.1X175 =85.5+17.5 =103 ns Multilevel Page Table (hierarchical structure) Large Computer System supports

of 1 KB will consist of one million (2 30 /2 10 =2 20) entries in the page table. Therefore, if entries are kept in a single table, will require a large amount of physical memory space. As memory is a precious resource its use must be optimized. The solution to the above problem is to divide this large single page table into multiple page tables. This lead to the development of the multiple page table/hierarchical structure. Fig. 10.7 Two level Page Table

Fig. 10.8 Address-Translation Scheme 10.5.2 Segmentation Segment is a logical entity about which the programmer is aware of A segment may contain a procedure or an array, stack or a collection of scalar variables. Usually, it does not contain a mixture of different type of entities. Each segment has a name and length. To specify address in this segmented memory or two- dimensional memory, the program need to supply a two-part address, a segment number and address within the segment called offset. For example, Intel 8086 microprocessor-based system has four segments namely code segment, data segment, stack segment and extra segment. The number of segments present in the system is computer architecture dependent. Some important properties of segments are- ➢ Each segment consists of a linear sequence of addresses, from zero to maximum. ➢ The length of each segment may be anything from 0 to maximum. ➢ Different segment may have different length. ➢ The segment length may change during execution. ➢ The OS maintains a free list and allocate segments to memory holes. ➢ Simplifies modification and recompilation of procedures and facilitate sharing of procedure and data. A segment table is very similar to the page table. A segment table entry must have fields for storing the segment's starting address in main memory and the size of the segment. If the maximum segment size is m bits, then last m bits of the logical address specify the segment offset. The remaining bits specify the segment number. The logical address is translated into a physical address by extracting the segment number and offset from the logical address.

is compared to the segment size. If the offset size is greater than the segment size, invalid address faults is generated, and abort the program. Otherwise, the offset is added to the segment's starting address to generate the physical address. To increase speed, the size check and physical

address generation can be performed concurrently. Fig. 10.9 Segmentation Because segments are user defined, it is possible to define certain segments to be read only. By adding a read only bit into a segment table entry, the memory management system can check for write operations to read only segments and generate a fault if such an operation is detected. Shared segments may also be used by processes executing different programs but using the same subroutine library. In this situation, care must be taken that addresses within the shared segment will work with both programs. For addresses to locations within the segment, the simplest solution is to use relative addressing. Relative addressing uses the offset from the current value of the program counter to generate the destination address. For all addresses, indirect addressing through a register pointing to the appropriate segment is also a possibility. Direct addressing, which specifies the segment and offset of the destination address, is possible only if all programs use the same segment number for the segment being accessed. 10.5.3 Segmentation with Paging Segmentation can be combined with paging to provide the efficiency of paging with the protection and sharing capabilities of segmentation. When paging is added in the segmentation, the segment offset

90%	**MATCHING BLOCK 158/158**	SA	OS_Book.pdf (D137278070)

is further divided into a page number and a page offset.

The segment table entry contains the address of the segment's page table. The hardware adds the logical address's page number bits to the page table address to locate the page table entry. The physical address is formed by appending the page offset to the page frame number specified in the
page
table entry.
Fig.10.10 Segmentation with paging 10.6
Demand Paging Demand
paging system is similar to a paging system with swapping. Processes reside onto secondary memory.
When a process is to be swapped in, the pager guesses which pages will be used before the process is swapped out again. Instead of swapping in a whole process, the pager brings only those necessary pages into memory. Thus, it avoids reading into memory pages that will not be used anyway, decreasing the swap time and amount of physical memory needed.
Demand paging is the concept of bringing the page in primary memory from secondary memory when it is needed. In demand paging: • Less I/O needed • Less memory needed • Faster response • More users can refer the page because page is referenced at the time of need • If page is not-in-memory then bring into memory
Fig. 10.11 Demand paging 10.7 Working Set Model The working set of a process at any particular time is the set of pages referenced over some preceding time interval. The working set can be expressed as W (t, ?) where W represents the working set, t represents the time and ? represents the interval. The goal is to pick a value for ? so that working set reflects the locality of the program. Too small a value for ? and the working set will not include all the pages in the current locality. Too large a value for ? and the working set includes pages from a previous locality. Knowledge of the size of the working set can be used to determine the number of pages to allocate to each process. 10.8 Prepaging In pure demand paging, large number of page faults occurs. This situation is a result of
trying to get the initial locality into memory. For example when a swapped out process is restarted, all its pages are on the disk must be brought in. Prepaging is an attempt to prevent this high level of initial
page faults. The concept is to bring into memory at one time all the pages that will be needed. In a system using the working set model, when the process is to be resumed, we automatically bring back into memory its entire working set before restarting the process.
Prepaging may be an advantage in some cases, but • Cost of prepaging may exceed the cost of servicing the page faults. • Many of the pages brought back into memory
may be of no use. • Sometimes it will be difficult to guess the pages which will be required. 10.9
Thrashing When
the number of frames allocated to a low priority process falls below the minimum number required by the computer architecture, we
need to
suspend that process's execution. We should then page out its remaining pages, freeing all its allocated frames. This introduces a
high amount of swap-in, swap-out of pages. Let we have a process that does not have enough frames, it will generate some page faults. At this time, some pages must be replaced.

However, since all its pages are in active use, it must replace a page that will be needed again right away. Consequently, it quickly faults again, and again, and again.

The process continues to fault, replacing pages for which it then faults and brings back in right away.

This high paging activity is called

thrashing.

A process is thrashing if it is spending more time

in paging then executing. 10.10 Page Replacement Algorithm When a reference is made to a page, not present in main memory is called page fault. When page fault occurs, it is necessary for OS to read in the required page from the secondary memory. Removing the page randomly will not be a good idea hence we need to have page replacement algorithm. In a computer operating system which utilizes paging for virtual memory management,

page replacement algorithms decide what pages to page out (

swap out) when a page needs to be allocated.

This happens when a page fault occurs and a free page cannot be used

to satisfy the allocation, either because there are none, or because

the number of free pages is lower than some threshold. When the page that was selected for replacement and paged out is referenced again it has to be paged in, and this usually involves waiting for I/O completion. This determines the quality of the page replacement algorithm: the less time wasted by waiting for page-ins, the better the algorithm. A page replacement algorithm

tries, by looking at the limited information about accesses to the pages provided by hardware, to guess what pages should be replaced in order

to minimize the total number of page misses, while balancing

this

with the costs (primary storage and processor time) of the algorithm itself. There are various page replacement algorithms

available. Some of the very important and common algorithms are discussed:

1. FIFO: The first in first out page replacement algorithm selects the page that has been in memory the longest. To implement this algorithm page table entries must include a field for the swap in time. When a page is swapped in the OS loads the field with current time. The page selected for replacement will be the one with the earliest swap in time. Although easy to implement and inexpensive FIFO is not very efficient. Frequently used pages even though they are in memory should not be swapped out. FIFO doesn't consider the amount they have been used and swaps them out anyway. Example: 2. LRU: The least recently used algorithm keep tracks of the last time each page was used not when it was swapped in. The memory management hardware uses a counter which is incremented during each memory reference. Each page table entry has a field that stores a value of this counter. When a page is referenced the value of the counter is updates in the page table entry for that page.

LRU replacement associates with each page the time of that page's last used. When a page must be replaced LRU chooses

that

page which

has not been used for the longest period of time.

This strategy is

the

optimal page replacement algorithm looking backward in time rather than forward.

3.

Second chance page replacement

algorithm: The basic algorithm of second chance replacement is a FIFO replacement algorithm. When page has been selected, we inspect its reference bits. If the value is 0, we proceed to replace this page

and

the reference bit is set to 1, however we give that page a second chance and move on to select the next

FIFO page. When a page gets a second chance its referenced bit is cleared and its arrival time is reset to the current time. Thus, a page that is given a second chance will not be replaced until all other pages

are replaced.

In addition, if a page is used often enough to keep its reference bit set, it will never be replaced.

Fig. 10.12 Second chance page replacement algorithm Summary Memory is an important resource, Memory Management Unit manage all functions related to memory management. Compiler binds symbolic addresses to relocatable addresses at various times like

Compile time, Load time, and Execution time. An address generated by the CPU (user programme) is referred as Logical Address

and an address seen by the memory unit is called as Physical Address. When a routine is loaded at the time of calling is known as Dynamic Loading. MMU of OS provide mechanism for memory allocation to the requesting processes. Common algorithms used for this purpose are First Fit,

Best Fit and Worst Fit. Memory allocation algorithms suffer from the problem of fragmentation like External Fragmentation and Internal Fragmentation. Combine all memory holes into one big hole by moving all the processes downward is known as memory compaction. The basic idea behind Virtual Memory is to execute the programs of size larger then physical memory available. Paging is a memory management scheme in which memories as well as processes are divided into fixed size of blocks. The address translation tables for mapping virtual address to physical address is known as a Page Table.

A special small fast lookup hardware cache called Translation Look-aside Buffer (TLB)

also known as Associative Memory.

The percentage of time that a particular page is fond in the TLB is called the Hit Ratio.

Multilevel Page Table or hierarchical structure is used to improve memory usage and is achieved by dividing large single page table into multiple page tables. Segment is a logical entity which is very similar to the page table. Segmentation can be combined with paging to provide the efficiency of paging with the protection and sharing. Working Set is the set of pages referenced over some preceding time interval. Prepaging means to bring all the pages that will be needed into memory at one time.

High paging activity is called

Thrashing,

a process is thrashing if it is spending more time

in paging then executing. There are various page replacement algorithms available like FIFO, LRU, Second chance page replacement etc. Questions 1. What are the four main functions of a memory manager? 2. Where is the MMU located? 3. What does the physical address space of a machine contain? 4. Why does FIFO have poor performance? 5. What is a downfall of memory compaction? 6. What does MMU stand for? 7. What is meant by SWAPPING OUT a process? 8. What is the aim of memory compaction? 9. What maps virtual pages onto physical pages? 10. What is a free list? 11. What is internal fragmentation? 12. What is external fragmentation 13. What is the method of relocation? 14. When does external fragmentation occur?

15. What is a page frame? 16. What is the purpose of a physical page? 17. A virtual address has two parts, what are they? 18. How is 'page size' determined? 19. In a paged system, how is an address viewed? 20. What is a page map? 21. Where is the information required to do mapping contained? 22. What is a page table? 23. What is the relation between page tables and processes? 24. What is contained within a page table entry? 25. What does the number of page table entries depend on? 26. What does the valid bit represent? 27. Describe the steps involved in a page table operation? 28. Who checks the valid bit and permissions of a requested memory access in a PTE? 29. What is a page fault? 30. Who throws a page_fault exception? 31. Explain why small page sizes necessitate keeping the page table in memory rather than in hardware registers. 32. What is the difference between external and internal fragmentation? 33. What is the difference between page and a page frame? 34. What are the problems with non-contiguous logical address space? 35. Do page and page frames have to be the same size? Why or why not? Further Readings 1. Modern Operating System – A.S. Tanenbaum (PHI) 2. Operating System Concepts – Silberschatz A. and Galvin Peter B. (Addision Wesley) 3. Operating Systems – Deitel H.M. (Addision Wesley) 4. Operating Systems – Stalling W. (PHI) 5. The design of the UNIX O.S. – Bach, M.J. (PHI)

UNIT 11: File System 11.0 Objectives 11.1 Introduction 11.2 File Concept 11.2.1 Components of a file 11.2.2 Logical data organization 11.3 Directories 11.4 Disk Organization 11.5 Disk Space Management 11.5.1 Linked List 11.5.2 Bit Map 11.6 Disk Allocation Methods 11.6.1 Contiguous allocation 11.6.2 Noncontiguous Allocation 11.7 Disk Access Scheduling 11.7.1 FCFS (First-Come-First-Served) Scheduling 11.7.2 SSTF (Shortest-Seek-Time-First) Scheduling 11.7.3. Scan Scheduling 11.8 File Protection, Access and Sharing 11.9 Introduction to Unix and Windows File System 11.9.1 Unix File System 11.9.2 Windows File System 11.10 Introduction to I/O sub system 11.11 Device Drivers 11.12 Terminal I/O 11.0 Objectives After going through this unit student will be able to: • To define

what is a file and directory. • Identify the disk space management methods. • Identify the disk accessing techniques. • File organization in different operating systems.

11.1 Introduction A file is a named collection of logically related information. The computers are becoming more and more capable and are able to perform complex functions on large amount of data. This data is stored in files. A file can contain any type of data like program, report, executable file, binary data, resume, presentations etc. The file system is the most visible part of any operating system. Users are always aware of the existence of files and their attributes like read, write or executable, time of its creation, owner, and size and so on. Files can also store a data collection which is too large to fit in the available main memory. A file is an important factor of information management of a computer system. 11.2 File Concept

File system is one of the most visible services of an operating system.

Files are normally stored on physical devices such as hard disk, floppy disk, magnetic tape, CD-Rom, DVD etc. where characteristics and organization are different from each other. While creating or manipulating files in any operating system environment, users need not be aware of physical characteristics of devices. File mapping is done by file management which is supposed to hide all device specific aspects of file manipulation from users and provide them with an abstract of a simple, uniform space of named files. File often appears as

100%	MATCHING BLOCK 156/158	W

a sequence of bits, bytes, lines or records whose meaning is defined by

a programmer or user. The common responsibilities of the file management system include the following: ➢ Mapping of access requests from logical to physical file-address space. ➢ Transmission of file elements between main and secondary storage. ➢ Management of the secondary storage, such as keeping track of the status, allocation and deallocation of space. ➢ Support for protection and sharing of files and the recovery and possible restoration of files after system crashes. 11.2.1 Components of a file i. Byte - A byte is a group of eight bits and is used to store a character. ii. Data Item(Field) – A data item is the smallest unit of named data. It may consist of any number of bits or bytes. A data item represents one type of information and is often referred to as a field or data element. iii. Record – A record is a group of named collection of data items or row which represents a

complete unit of information. A logical record contains contents of all the fields. Therefore, a file contains Records, Records contain fields, Fields contain data items which contains any character. Each character occupies one byte for its storage. 11.2.2 Logical data organization Logical data organization refers to the manner in which data are represented to the application programmer or user of data. The terms a programmer is normally used while defining the structure of a file are: Field Length A field is a data element which can store one type of information which is composed of characters, numbers, special characters etc. that a field can store is called field length. As each character requires one byte for its storage, number of characters inside a field determines its field length in bytes. There can be fixed length fields as well as variable length fields. Record Length A record is a collection of fields. Total sum of field lengths of all its fields is called Record Length. Even records can be of fixed and variable length. 11.3

Directories In many systems, directories are treated as files which keep track of all other files. A single level flat directory can contain a list of all files in a system. Flat directories are those directories in which the root directory contains all system files and there is no other subdirectory. Flat directories are inconvenient and often inadequate for large systems with many files and possibly multiple users because the list of all files may be too big to display in one screen of text. Another

Fl-21 Fl-11 Fl-22 Dir-31 FL-41 Dir-21 FL-311 Hierarchical directory structure DIR-52 DIR-51 Dir-11 Dir - 5 Dir - 4 Dir - 3 Dir-2 Dir-1 Root Directory

problem with flat directories in larger system includes difficulties in unique naming of files and inadequate support for selective sharing and protection of files in multi-user systems. In hierarchical directories (i.e. a tree of directories), the collection of all directory and subdirectory entries define the totality of system files. This is the more natural way to organize files. A user can group the related files into subdirectories. All system files are then cataloged in a hierarchy of directories whose structure is similar to that of a rooted tree. By associating different access rights with different subdirectories that need to be referenced when accessing a given file, selective sharing and protection can be easily supported in systems with hierarchical directories. This system is represented in the following figure:

The advantages of hierarchical directories are: • Facility of unique naming of files • Support for selective sharing and protection • Convenient directory manipulation There are two basic methods to access a file organized in a hierarchical form. In the first method, each file is given an absolute pathname consisting of the path from the root directory to the files. For example /dir-3/dir-31/ fl-311 means that the root directory contains a subdirectory dir-31 which in turn contains a file FL-311. Absolute path always starts from the root directory and are unique. The other method is the relative path name. This is used in conjunction with the concept of the

working directory i.e. current directory. The most common operations on directories are searching, addition and removal of entries. Since directories are usually stored on the disks, the efficiency of directory operation is primarily affected by the numbers of disk access required to complete a search, to append or to remove an entry. Therefore, directories are organized in such a way that these operations should take less time. Directories are basically symbol tables of files. An entry of a directory defines a file. A file is usually defined by its name, its attributes and an access pointer. A typical directory entry may contain the following information: • File name, type and version number • File-access pointer (starting disk address) • File attributes: o Size o Structure o Owner o Access rights o Date of creation o Date of last backup o Date of last reference Directory entries are added by the creation of files and of aliases to existing files. Entries are removed from directories when files are deleted. Directories are frequently searched to locate and access files for processing and to add or to remove entries. One of the concerns is the location of file directories. Should directories be kept on disk or in memory? As average disk-access times indicate much faster access to and processing of directories may be accomplished by keeping them in memory. On the other hand, space needed for storage of directories can become quite large, so that directories should be kept on secondary storage. The efficiency of directory operations is therefore primarily affected by the number of disk accesses required to complete a search or to append or remove an entry. A linear list organization of directories is not very suitable for searching since it takes $O(n)$ (Order of n) comparisons to locate a given entry where n is the number of all entries in a directory.

Hashing technique supports fast searching and appending operation but in the worst behavior of hashing it is no better than that of linear list. Moreover, when a directory exceeds the limit that the related hash function can address, all of its entries may be hashed again. Such problems may be circumvented by using tree structures to represent directories. A balanced binary tree may be searched in $O(\log n)$ time. Appending and removal of entries can also be performed in this time. An additional advantage of trees is that they facilitate sorted listing of directory entries, which is required by many systems. 11.4

Disk Organization A variety of input/output devices can be used for file storage in a computer system. Their general characteristic is the addressability and transmission of data in blocks. Information on the disk is referenced by a multi-part address, which includes the drive number, the surface and the track. The collection of tracks on all surfaces that are at the same distance is called a cylinder. A simplified physical organization of a magnetic disk is presented in the following figure: The medium for data storage is the magnetic oxide coating, a disk platter which resembles a phonographic record. One or more such platters may be available for data storage and retrieval in a single disk drive.. Disk platters are constantly rotated by the drive mechanism at a speed of 3000 rpm or higher, floppy disks rotate at about 300 rpm and they can be stopped completely between accesses. Data are read and written by means of the read/write heads mounted on a head assembly in such a way that they can be brought in close contact with the portion of disk where the target data reside. Data are stored on the magnetic disk surface in the form of concentric circles called tracks. The collection of tracks on all surfaces that are the same distance from the disk spindle is called a cylinder. For example, as per the above figure, disk has two platters and four recording surfaces (two on each platter). Thus, each of its cylinders consists of four tracks. The number of distinct tracks on a single surface determines the total number of cylinders on a given disk. Usually, a number of blocks of data, called sectors are recorded on each track. Disk Access Time Depending on the number of the available read/write heads, disks are either of the fixed head or of the moving-head variety. Fixed head disks usually have a separate read/write head for each track. A given sector is accessed by activating the head on the appropriate track when the target sector passes under it. The time necessary to access the desired sector is called the rotational latency. On an average, it equals one-half of the disk revolution time, which is on the order of milliseconds for typical disk rotational speeds. Moving-head disks are characterized by having either one or just a few read/write heads per surface. Removable disks are usually of the moving-head variety in order to allow the head assembly to retract from the cartridge before it is replaced. Reading of a sector requires that the head assembly first moved to the corresponding cylinder. Then the head on the related track is activated when the target sector passes under it. Thus, the access time of a moving head disk includes both the head positioning time or seek time and the rotational latency. Hardware-related delays in transferring data between disk and memory are a combination of three primary factors: ➢ Seek time – the time necessary for the read/write heads to travel to the target cylinder. ➢ Rotational latency – the time spent waiting for the target sector to appear under the read/write heads. ➢ Transfer time – the time necessary to transfer a sector between the disk and the memory buffer. 11.5 Disk Space Management An important function of the file system is to manage space on the secondary storage. This includes keeping track of both disk blocks allocated to files and the free blocks available for allocation. In addition to directories that describe existing files, the file system must maintain a pool of free blocks. Whenever a file has to be created, the list of free disk space is searched for and then allocated to the new file.

The amount of space allocated to this file is then removed from the free space list. When a file is deleted, its disk space is added to the free space list.

A good space allocation strategy must take into consideration several related and interactive factors such as: • Processing speed of files and allocation and deallocation of blocks. • Ability to make use of multi sector and multi track transfers • Disk space utilization • Main memory requirements of a given algorithm The two methods used to manage free disk blocks are explained below: 11.5.1 Linked List In this method all free disk blocks are linked together by each free block pointing to the next free block. There must be another pointer pointing to the first free block of a linked list which will be pointing to the second free block which would further point to the third free

block and so on.

This scheme is not very efficient since to traverse the list, we must read each block requiring substantial time. Another disadvantage is additional memory requirement for maintaining linked list of all free disk blocks. 11.5.2 Bit Map Usually the list of free disk spaces

is implemented as a bit map or bit vector. Each block is represented by

a single bit. Zero(0) is marked as a free block where as One(1) is for allocated block. Advantage of this approach is that it is a simple method and efficient to find n consecutive free blocks on the disk, but the disadvantage is that extra disk space is required to store bit map. A disk with n blocks requires a bit map of n bits. As compared to linked list the bit map requires less space since it uses 1 bit per block versus 16 bits in the liked list. 11.6

Disk Allocation Methods Management of storage blocks is a familiar problem in relation to main memory management. An important function of the file system is to manage space on the secondary storage, which includes keeping track of both disk blocks allocated to files and the free blocks available for allocation. The main problems in allocating space to files are: • Effective utilization of disk space • Fast accessing of files. Secondary storage introduces two additional problems: • Slow disk access time and • Larger number of blocks to deal with Many considerations are similar to both environments, particularly contiguous and non contiguous allocation of files. Three widely used allocation techniques are: • Contiguous allocation • Linked (Non Contiguous allocation) • Indexed (Non Contiguous allocation) To study the above allocation strategies, we will consider that a file is a sequence of blocks. All I/O operations on disk occur in terms of blocks. 11.6.1 Contiguous allocation The basic idea of contiguous allocation of disk space is to allocate contiguous areas of disk in response to run-time requests. A user specifies in advance the size of the area needed to hold a file to be created. If the desired amount of contiguous space is not available, the file cannot be created. The advantage of contiguous allocation is that all successive records of a file are normally physically adjacent to each other. This increases the accessing speed of records. It means that if records are scattered through the disk its accessing will be slower. For sequential access the file system remembers the disk address of the last block and when necessary reads the next block. Random access to

contiguous files is also fast, because the address of the target disk block can be easily calculated on the basis of the file's starting address recorded in the directory. The difficulty with contiguous allocation is how to find a space for a new file, if the implementation of a free space list is done through a bit map method then for the creation of a n- bytes long file, we need to find n zero(0) bits in a row. The main problem with contiguous allocation is the disk fragmentation that it produces. Due to the variability of file sizes, disk space can become heavily fragmented. A combination of internal and external fragmentation may cripple disk utilization. A form of internal fragmentation is present in all file systems, regardless of the space allocation policy in use. The most common strategies for allocation are: i. First – fit :- In this case as soon as the first hole(big enough space) is encountered, searching is stopped and memory is allocated for creating a file.

Searching can start either at the beginning of the set of holes or from where the previous first-fit search ended. ii. Best – Fit :- In this method, the entire list is searched for and the smallest hole, that is big enough, is allocated for creating a file. By comparing the both method, we concludes that

first fit is generally faster. These algorithms suffer from external fragmentations. As files are allocated and deleted the free disk space is broken into little pieces. External fragmentations

refer to the scattered group of free blocks that are too small for allocation but which as a collection may represent a large percentage of disk size. Contiguous allocation of disk space requires the file size to be preclaimed that is, estimated, at the time of its creation. Although this is not a problem for copying files, accurate estimation of file sizes is generally difficult and unreliable.

11.6.2 Noncontiguous Allocation Noncontiguous allocation of disk space, in terms of both files and free space tends to grow or shrink overtime and because users rarely know in advance how large their files will be contiguous. The following two strategies are used in Noncontiguous allocation:- • Chaining (Linked) Allocation • Indexed Allocation Chaining Chained allocation is essentially a disk based version of the linked list. In linked list

allocation each file is linked list of disk blocks. These blocks may be scattered through the disk.

In case of files, the directory may contain the "head pointer", that is the address of the first block. A few bytes of each disk block contain the address of the next block. Chained files are well suited for sequential access because the block being processed contains the address of the next block in line. The advantage of chained allocation is: i. Its simplicity ii. No disk compaction required. Due to noncontiguous nature of allocation, the linking does not produce any external disk fragmentation. Any disk

block on the free space can be used to satisfy a request,

since all blocks are linked together. There is also no need of declaration of

the size of a file in linked allocation while it is created. A file can continue to grow as there are free blocks.

The disadvantages of chained allocation are: i. Slow direct accessing of any disk block. ii. Space requirement for pointers iii. Reliability – as disk blocks are linked by pointers, a single damaged pointer can make lot of disk blocks inaccessible. Indexed Allocation Indexed allocation support direct accessing by placing all of the pointers together into a index block which is not supported by chained allocation. Although similar to chaining in that it also maintains pointers to allocated blocks, indexing attempts to improve the speed of random access by collecting pointers into the so-called index blocks. Each file is provided with

100%	**MATCHING BLOCK 157/158**	W

its own index block, which is an array of disk block

pointers (addresses).

The directory contains the address of the index block. Although quite different in their implementation index blocks basically serve the same purpose as page map table does in paged memory systems. The advantage of indexed file allocation includes the absence of external fragmentation and the efficiency of random accessing. Moreover, indexing can easily map around the bad disk blocks. Indexing of free space can be accomplished by means of the bit map. The primary disadvantage of indexing is the number of disk accesses necessary to retrieve the address of the target block on disk. This overhead can be reduced by keeping some index blocks in memory. The other disadvantage is that indexed allocation requires lots of space for keeping pointers.

The pointer overhead of the index block is generally worse than the pointer overhead of linked allocation.

Indexed allocation is more complex. If the index block is already in memory, then the access can be made directly. However, keeping the index block in memory requires considerable space.

Some systems combine contiguous allocation with indexed allocation by using contiguous allocation for small files and automatically switch to an indexed allocation if the file grows large. Since most files are small and contiguous allocation is efficient for small files, average performance can be quite good. 11.7

Disk Access Scheduling The performance of a computer system is dependent upon how fast a disk access request is serviced. To service a request, a disk system requires that the head be moved to the desired track, then a wait for latency and finally the transfer of data. The following scheduling algorithms are used to service the access requests. 11.7.1 FCFS (First-Come-First-Served) Scheduling This is the simplest form of disk scheduling in which the first request to arrive is the first one serviced. FCFS has a fair policy in the sense that once a request has arrived, its place in the schedule is fixed irrespective of arrival of a higher policy request. FCFS is easy to program, however, it may not provide the best service. 11.7.2 SSTF (Shortest-Seek-Time-First) Scheduling In SSTF scheduling priority is given to those processes which need the shortest seek, even if these requests are not the first ones in the queue. It means that all requests near to the current head positions are serviced together before moving head to distant tracks. 11.7.3. Scan Scheduling This scheduling strategy was developed to overcome the discrimination as well as to recognize the dynamic nature of the request queue. The read/write head of a disk starts from one end and move towards

the other end, services requests as it reaches each track until it reaches to another end of the disk.

After reaching another end of disk, disk head reverses its path-direction while continuing with services whichever comes on the way. This way disk head continuously oscillates from end to end. 11.8 File Protection, Access and Sharing The original motivation for protection mechanisms came with the advent of multiprogramming. The intent was to confine each user's program to its assigned area of memory and thus prevent programs from harming each other. The protection of a file is mostly needed in a multi-user environment where a file is shared among several users. Protection mechanism must provide controlled access by restricting the types of files which can made. Several different types of operations may be controlled. Different operations which can be performed on files are: • Reading • Writing • Executing • Appending • Deleting These operations can be represented by Access-Matrix model of protection as under: User/Object File1 File2 File3 Printer User1 Read, Write Execute Print User2 Read Write User3 Read, Write, Execute, copy Print In the above small access matrix, blank entries indicate no access rights. Although a useful model, access matrices are inefficient for storage of access rights in a computer system because they tend to be large and sparse. The actual form of representation differs in practice in accordance with the access-control mechanism in use. The common access-control

mechanisms for protection are: • Access hierarchies such as levels of execution privilege and block structured programming languages. • Access Lists of all subjects having access rights to a particular object. • Capabilities or tickets for objects that specify all access rights of a particular subject. • Password is associated with each file to control the access. • Access groups are used

to condense the length of the access list, many systems recognize three classifications of users in connection with each file: ➢ Owner – The user who created the file. ➢ Group – A set of users who are sharing the file

and need similar access. ➢ Universe – All other users in the system.

Many different protection mechanisms have been proposed. Each one has got advantages and disadvantages and must be selected which is suitable for its intended application. On the basis of rights, users can share and access the files. 11.9 Introduction to Unix and Windows File System 11.9.1 Unix File System Probably the simplest definition of a file system is a data structure or a collection of files. In UNIX, file system can refer to two very distinct things; the directory tree or the arrangement of files on disk partitions. The latter can be thought of as the physical file system as it has a tangible physical location. The logical file system refers to a hierarchy of connected directories made of all the files (or disk partitions) that are accessible to the user. The UNIX file system is arranged in a tree or inverted pyramid, where all files are logically contained in the root file. In UNIX, the files are organized into a tree structure with a root named by the character '/'. The first few levels of the tree look like this:

There are four types of files in the UNIX file system. Ordinary Files An ordinary file may contain text, a program, or other data. It can be either an ASCII file, with each of its bytes being in the numerical range 0 to 127, i.e. in the 7-bit range, or a binary file, whose bytes can be of all possible values 0 to 255, in the 8-bit range. Directory Files Suppose that in the directory x I have a, b and c, and that b is a directory, containing files u and v. Then b can be viewed not only as a directory, containing further files, but also as a file itself. The file b consists of information about the directory b; i.e. the file b has information stating that the directory b has files u and v, how large they are, when they were last modified, etc. Device Files In Unix, physical devices (printers, terminals etc.) are represented as "files." This seems odd at first, but it really makes sense: This way, the same read() and write() functions used to read and write real files can also be used to read from and write to these devices. Link

File

Suppose we have a file X, and type ln X Y If we then run ls, it will appear that a new file, Y, has been created, as a copy of X, as if we had typed cp X Y However, the difference is the cp does create a new file, while ln merely gives an alternate name to an old file. If we make Y using ln, then Y is merely a new name for the same physical file X.

Roo t etc bin usr tmp dev ls csh ucb lib

Obtaining Information About the Files in a Given Directory The 'a' ("all") and 'l' ("long") options of the ls command will give us a lot of information about files in a specified directory (if we don't specify a directory, then the current directory is assumed). Here is a sample output from typing ls -al drwxr-xr-x 6 ecs4005 1024 Apr 22 13:30./ drwxr-xr-x 74 root 1536 Mar 24 12:51../ -rw——— 1 ecs4005 188 Apr 13 15:53.login -rw——— 1 ecs4005 6 Mar 24 11:29.logout -rw——— 1 ecs4005 253 Apr 10 12:50.xinitrc -rw-r—r— 1 ecs4005 516 Apr 10 13:00.twmrc -rw-r—r— 1 ecs4005 1600 Apr 22 10:59 test2.out The output is separated into six columns: 1st column - access permissions (see below) 2nd column - number of file entries (in the case of directory files) 3rd column - owner 4th column - size in bytes 5th column - date and time of last modification 6th column - name File Access Control In Unix, all files are protected under some access control mechanism, so that the owner of a file can deny access of his files to other users. The first column of the long directory list shows the access characteristics of a file, in the form of 10 flags, e.g. drwxr-xr-x. The meanings of the flags are shown below: Position 1 file type: d (directory) - (ordinary file) l (symbolic link) Position 2-4 permissions for the owner: r (read) w (write) x (execute) Position 5-7 permissions for other users in the same group Position 8-10 permissions for all other users

Note that a hyphen ('-') denotes lack of the given permission type. For example, r-x would mean that read and execute permission are granted, but not write permission. In order to remove a file, you must have write permission for it. In order to view the contents of a directory, i.e. see what files are there, you need read permission for that directory. In order to actually access a file (read from it, write to it, or execute it) in the directory, you need execute permission for the directory. i-nodes A Unix file is described by an information block called an i-node. There is an i-node on disc for every file on the disc and there is also a copy in kernel memory for every open file. All the information about a file, other than it's name, is stored in the i-node. This information includes • File access and type information, collectively known as the i-node. • File ownership information. • Time stamps for last modification, last access and last mode modification. • Link count. • File size in bytes. • Addresses of physical blocks. 11.9.2 Windows File System Three local file systems currently supported on the Microsoft® Windows® 98, Microsoft Windows XP, Microsoft Windows 2000, Microsoft Windows Server™ 2003, and Microsoft Windows Vista platforms: • File Allocation Table (FAT) • NTFS • Universal Disk Format (UDF) FAT The FAT file system originated in the late 1970s and early1980s and was the file system supported by the Microsoft MS-DOS® operating system. It was originally developed as a simple file system suitable for floppy disk drives less than 500 KB in size. Over time it has been enhanced to support larger and larger media. Currently there are three FAT file system types: FAT12, FAT16, and FAT32. The basic difference in these FAT sub types, and the reason for the names, is the size (in bits) of the FAT entries in the actual FAT structure on the disk.

The FAT file system is ideal for removable Flash media that is used in consumer electronic devices, such as digital cameras, media players, Flash drives, and so on. FAT12 is, of course, the only file system that can be used on floppy disks. FAT 12 and FAT 16 are recognized by all operating systems from MS-DOS onward. FAT 32 is recognized by all operating systems from Windows 98 onward. NTFS Microsoft designed the Microsoft Windows NT® File System (NTFS), specifically for the Windows NT operating system, to support and include features required from an enterprise-class file system. Most PCs are configured with fixed hard drives. These PCs are used for a variety of purposes, including file share servers, mail servers, database servers, and home or information worker desktops. There are many requirements for a file system that arise from these different usage scenarios, such as recoverability, availability, security, and performance. NTFS meets all of these requirements, which makes NTFS is the ideal file system for use with fixed hard drives. UDF UDF is a file system that is defined by the Optical Storage Technology Association (OSTA). UDF is compliant with ISO-13346/ECMA-167 and is the successor to the CD-ROM file system (CDFS or ISO-9660). UDF can also be used for exchanging data among computers running different operating systems. The UDF format supports a number of advanced features including; long and Unicode filenames, access control lists (ACL)s, alternate data streams, 64-bit file sizes, sparse files, and writing to many different types of media. UDF on CD/DVD media is ideal for storing media files. Because of the storage capacity of DVD media types, DVD media with UDF is replacing tape as the best choice for archiving data 11.10 Introduction to I/O Sub System The input/output system is the part of the operating system charged with management of I/O devices. There is a considerable variety of device types attached to computer systems. Many of them operate asynchronously from the processor, at rates ranging from just a few data transfers per second to millions of data transfers per second. The I/O system acts as an interface between programmers and I/O devices.

In general, most I/O devices share two important characteristics that largely define the input/output problem:- • Asynchronous operation • The speed gap Asynchronous operation During program execution, the processor engages in numerous transactions with the main memory and I/ O devices in order to fetch instructions for execution and to access data operands. For example, READ operations with both memory and devices are typically initiated by the processor, which some time later, obtains the required items. However, the timing of memory and that of I/O transactions are fundamentally different. From the point of view of the processor, I/O data arrival and transfer times are generally unpredictable, once the processor informs the terminal controller that it needs a character, the actual data input cannot take place until the user at the terminal presses a key. That may happen a fraction of a second or minutes later. Thus I/O devices, unlike memory, operate asynchronously from the processor. The Speed Gap: Processor versus peripherals There is a difference of several orders of magnitude in the speeds of I/O devices themselves, and of I/O speed in general, relative to the processor memory bandwidth. Very high speed devices are usually interfaced via specialized I/O channels or dedicated processors. They tend to perform data transfers in blocks using direct memory-access (DMA) techniques. In contrast, programmed I/O, which is controlled directly by the processor for each individual character, is often limited to the less demanding devices. INPUT/OUTPUT Interfaces Hardware controllers (interfaces) act as intermediaries between the computer system and input/output devices. Their purpose is to bridge the speed and signaling level incompatibilities between the processor and peripherals, and to translate the generi c I/O commands issued by the processor into device-specific controls. I/O hardware is often physically distributed over a number of device interfaces or controllers. This is shown in the following figure:-

Block Structure of a computer system As their respective labels indicate, the I/O controllers are in charge of the serial, parallel and DMA devices. Many terminals, as well as short and long haul communication links use bit-serial line protocols and thus require serial interfaces for connection to a computer system. The majority of printers need parallel interfaces. The following figure shows a very general block diagram of an I/O controller:- Block structure of an I/O Controller The controller is divided into three functional layers: • Bus interface • Generic device controller • Device interface The following figure shows the block structure of a typical I/O port:

Block structure of an I/O port The port is assumed to be bidirectional i.e. capable of simultaneous data transmission in both input and output directions. As per the above figure, the port is connected directly to the computer system bus and to the I/O device at its respective ends. The most important part of the I/O port's structure is the set of registers that represents the programmer's view of the attached I/O device. The four register types shown in the above figure represents the major functional categories commonly found in I/O ports: • Data registers (input and output buffers) • Command registers • Status registers Buffer registers are generally used to hold the data in transit until both I/O and processor are ready and willing to conclude the I/O transactions. Command and status registers are parts of the handshaking mechanism used to carry out the negotiation protocol and momentarily synchronize processor with I/O devices in order to exchange data. 11.11 Device Drivers The purpose of a device driver is to handle requests made by the kernel with regard to a particular type of device. There is a well-defined and consistent interface for the kernel to make these requests. By isolating device-specific code in device drivers and by having a consistent interface to the kernel, adding a new device is easier.

A device driver is a software module that resides within the kernel and is the software interface to a hardware device or devices. A hardware device is a peripheral, such as a disk controller, tape controller, or network controller device. In general, there is one device driver for each type of hardware device. Device drivers can be classified as: • Block device drivers • Character device drivers (including terminal drivers) • Network device drivers • Pseudo device drivers Device driver configuration consists of the tasks necessary to incorporate device drivers into the kernel to make them available to system management and other utilities. 11.12 Terminal I/O The terminal device driver controls a lot more than just terminals. modems, printers, direct connections to other computers, and other special devices that rely on streams of characters. Terminal devices can be put into different states. The default state of the terminal when running in a shell is canonical mode, also known as cooked mode. In this mode, the terminal driver returns one line of data at a time from the terminal device. Any special characters are processed as they come into the device (^C, ^Z, etc.). The second state the terminal can be in is noncanonical mode, or raw mode. In this state, the terminal device driver returns one character at a time without assembling lines of data. Also, special characters are not processed in this mode. Programs such as vi, use this mode for data input and output. This allows complete control of input and output characters. A third state, one which Posix.1 defines is the cbreak mode. This mode is similar to raw mode, except that the processing of special characters still takes place and the corresponding signals are raised for the special characters. Summary A file is a named collection of logically related information.

File system is one of the most visible services of an operating system.

Files are normally stored on physical devices such as hard disk, floppy disk, magnetic tape, CD-Rom, DVD etc. In many systems, directories are treated as files which keep track of all other files.

Disk is widely used by operating system for I/O purpose. A disk normally contains device directory that contains files on the disk. There are two different ways of allocation of files: contiguous and non-contiguous (Linked and Chained). Contiguous allocation can suffer from external fragmentation. Linked allocation does not support direct accessing of files. Arranging pending disk requests to minimize seeks is called disk scheduling. In the end various ways for file protection and I/O subsystems are explained. Glossary Directory which keep track of all other files and subdirectories. Field A data item is the smallest unit of named data File is a named collection of logically related information. i-node A Unix file is described by an information block. Record A record is a group of named collection of data items or row which represents a complete unit of information. Rotational the time spent waiting for the target sector to appear under the read/write latency heads. Seek time the time necessary for the read/write heads to travel to the target cylinder. Transfer time the time necessary to transfer a sector between the disk and the memory buffer. Further Readings 1. Operating Systems – Concepts and Design by Milan Milenkovic, Tata MacGraw-Hill 2. Operating Systems Design and Implementation by Andrew S. Tanenbaum, Prentice Hall of India

Questions 1. What is disk scheduling? What are the main objectives of disk scheduling? 2. What are the disadvantages with linked list being used for disk space management? 3. What are the desirable characteristics of disk allocation? 4. What is i-node?

UNIT-12: Dos Operating System 12.0 Objectives 12.1 Introduction 12.2 History and Versions of MS-DOS 12.3 Physical Structure of Disk 12.4 DOS Files System 12.5 File Allocation Table (FAT) 12.6 Booting Process 12.7 The Command Prompt 12.8 Typing a Command 12.9 Drive Name 12.10 File 12.11 Directory 12.12 Files and File Name 12.13 DOS Commands 12.13.1 Internal DOS Commands 12.13.2 External DOS Commands 12.13.3 Wildcard Characters 12.14 File Management Commands 12.15 General Commands 12.16 Disk Management Commands 12.17 DOS Utility Commands 12.18 Using copycon to build a File 12.0

Objectives After going through this unit, you will be able to: • Describe Dos Operating System, History and version of DOS. • Describe

Booting Sequence, Dos System file, File, Directories etc. • Describe Structure of disk, FAT, internal and external dos commands. • Describe File management commands. Dos utility commands, etc.

12.1 Introduction: DOS is a collection of programs that permits a computer to supervise its own operations automatically or under operator control calling in program routines, languages & data is required. DOS was invented by M/S MicroSoft company. That is why it is also known as MS-DOS. There are many operating systems (OS) for the PC but MS DOS was one of the most popular OS. Nowadays, it has become part of Windows OS. MS-DOS performs the following tasks 1. Makes the computer easy to use with the help of this, a user can access peripheral devices such as printers, diskette drives, and fixed disk drives etc with the computer. 2. Makes it easy for you to use application and create and manages files on your computer. 3. Controls the way the computer uses its resources like the memory, printer, disk and CPU etc. 4. Controls the movement of information on the computer 5. Provide protection to program and data from being accidentally changed. 6. Works as interface between user & hard disk. 12.2 History and Versions of

MS-DOS DOS is an acronym for Disk Operating System.

When IBM released the Personal Computer in 1981. Microsoft created MS DOS, the Microsoft Disk Operating System for the IBM personal computers. It controls different parts of the personal computer i.e. the Input / Output devices like monitors, printers, disk drives etc. DOS is a single user and single tasking system. The first software was called as DOS Version1.0 which had all the basic features. Since then, several new features have been incorporated. The latest version of DOS is Ver 7.0. The basic hardware requirement for Version 7.0 is a 16 - bit Personal Computer with 640 KB RAM and a hard disk & Floppy drives. The basic software which makes the user to interact with the machine is the Operating System. Various commonly used versions of DOS are : 1. MS-Dos1.25 2. MS-Dos 2.11 3. MS-Dos 2.25 4. MS-Dos 3.0 5. MS-Dos 3.1 6. MS-Dos 3.2 7. MS-Dos 3.3 8. MS-Dos 4.0

9. MS-Dos 6.0 10. MS-Dos 6.2 11. MS-Dos 6.22 12. MS-Dos 7.0 12.3 Physical Structure of Disk The figure below shows physical structure of a floppy disk that is used to store files of the system. Hard disk contains a number of disk platters similar to the one shown in figure. The platter rotates continuously. The head of disk is a small coil that reads or writes on the position of the disk rotating below it. The data is stored in concentric set of rings. These are called tracks. The width of a track is equal to the width of the head. To minimize the interference of magnetic fields and to minimize the errors of misalignment of head, the adjacent tracks are separated by inter track gaps. As we go towards the outer tracks the size of a track increase but to simplify electronics same numbers of bits are stored on each track. The capacity of the disk is measured in BYTES (8 bits). The data is transferred from and to the disks in blocks. Block is a section of disk data and is normally equal to a sector (512 bytes). A track is divided into 10-100 sectors and these sectors should be either fixed or variable length sectors. Two adjacent sectors are separated by intra-track gaps. This helps in reducing the precision requirements of sectors. To identify the sector position normally there may be a starting point on a track or a starting and end point of each sector. 12.4 DOS

System Files The DOS operating system was available in 3-5

high density floppy disks. It contains backup utilities and disk compression drivers. However, all the stuff that goes into C:\DOS and its subdirectories are programs and utilities. The core DOS operating system consists of six files: ➢ The boot sector is a 512 byte record placed at the beginning of the C: drive when DOS was

installed, or placed there subsequently using the "sys c:" command. ➢ Two "hidden" files are stored in the root directory of the C: drive. They do not show up in a DIR listing unless the /A switch is used. On IBM PC DOS systems, they are IBMBIO.SYS and IBMDOS.SYS. On MS DOS systems, they are called IO.SYS and MSDOS.SYS. These files form the kernel of the DOS system. ➢ COMMAND.COM is the "shell" or command interpreter. It prints out the "C:\> " prompt and reads user commands. It also supports BAT files. ➢ The user configuration files are CONFIG.SYS and AUTOEXEC.BAT. The reader is assumed to be familiar with these files.

COMMAND.COM is initially stored in the C:\ root directory. The problem is that OS/2 and Windows NT have their own versions of COMMAND.COM. To avoid confusion, each COMMAND.COM should be stored in the subdirectory that belongs to its particular operating system. In normal use, this

means that the DOS version should be in C:\DOS. To relocate it, two statements must be added to the user configuration files: "SHELL=C:\DOS\COMMAND.COM" is added to CONFIG.SYS. "SET COMSPEC=C:\DOS\COMMAND.COM" is added to AUTOEXEC.BAT. The hidden files IBMBIO, IBMDOS, IO, or MSDOS have names that do not conflict with each other or with any system file belonging to any other operating system. They can stay in the C:\ root directory no matter what gets added to the system. This means that the volatile part of the DOS system consists of the boot record, C:\CONFIG..SYS, and C:\AUTOEXEC.BAT. 12.5

File Allocation Table (FAT) The file allocation table (FAT) is a table of entries describing how each cluster is allocated on the disk. Each cluster on disk has a entry. The FAT is simply a table of numbers. The entry in FAT indicates whether the corresponding cluster is in use or not. It holds a pointer to the next cluster. Thus FAT is a "reservation clerk" keeping record of the disk space. This is usually stored in the zero sector of the disk. The FAT of a computer disk tells about filewise usage of sectors. The FAT (also know as FAT, FAT16, and FAT32), is located in the master boot sector (first 63) of a bootable floppy disk or hard disk. You visually cannot see this. The FAT follows the boot sector. The FAT file system revolves around the File Allocation Table for which it is named. Each logical volume has its own FAT, which serves two important functions: it contains the allocation information for each

file on the volume in the form of linked lists of allocation units and it indicates which allocation units are free for assignment to a file that is being created or extended. The File Allocation Table (FAT) was designed and coded in Feb., 1976 by a kid named Bill Gates during a five day stay at the Hilton Hotel in Albuquerque. He developed it for a version of Basic that could store programs and data on floppy disks. The FAT design was incorporated by Tim Patterson in an early version of an operating system for the Intel 8086 chip. Gates bought the rights to the system, then rewrote it to create the first version of DOS. As a direct result, Gates is the richest man in America. The FAT file system is simple and reliable. It does not lose data even if computer crashed in the middle of an update. It does not use a lot of memory. It does, however, do a lot of extra administrative I/O to different areas of the partition. The directory is allocated at the start of the partition and it contains the table of free space. To write a new data set, or to add data to an old one, the disk arm must be constantly moved between the location of the directory and the place where the data is being written. Without optimization, a file can end up fragmented into many small pieces. When the system crashes, no data is lost. However, a FAT system may have removed disk area from the chain of free space but may not have yet assigned it to any permanent new data set. The CHKDSK or SCANDISK utility examines the FAT table to determine the status of every record on disk. The records which are not part of any data set may be returned to the free space chain. After CHKDSK finds unallocated sectors, it asks you whether they should be turned into files. If you were in the middle of creating a new file and the system crashed, and the data that was lost is extremely valuable, you might answer Yes. If you do answer Yes you can scan the recovered file scraps for the information you lost. Otherwise, answer No and the unallocated space will be recycled as free space. If the FAT table (Table 1) becomes corrupted, or what is known as disk errors, there is a second (backup) table (Table 2) available for programs like Scandisk uses to fix the FAT (Table 1). A corrupt FAT can cause you to get the message "No Bootable Media". At this point if Scandisk cannot fix it you will need to format and start over. It is recommended that you Low Level Format and reinstall Windows. If the Windows Setup program complains about the drive, replace the drive with a new one. By design FAT supports a maximum of 64K allocation units. When the disk partition is 32 megabytes

or less, an allocation unit is a 512 byte sector. However, as the disk gets larger, the units get larger. A 64 megabyte disk partition has 1K allocation units. A 128 meg partition has 2K units. A 256 meg partition has 4K allocation units. Each file occupies one or more allocation unit. As the allocation units get larger, any large number of small files wastes a lot of disk space. The classical FAT directory structure (before NT and Windows 95) limits file names to eight characters with a three character extension (as in the name "AUTOEXEC.BAT"). This 8.3 naming convention was borrowed from earlier DEC minicomputers. The FAT structure also maintains for each file a set of attributes (is it a System data set, should it be Hidden in the DIR display, should it be Archived next time the disk is backed up, is it Read-only). There is also a data and time stamp when the file was last changed. OS/2 allows a FAT file to have additional Extended Attributes. Since there is no room for these attributes in the FAT directory, OS/2 creates a separate hidden file on the disk volume named "EA DATA. SF" and stores the information there. FAT has been around for a long time, so its problems have been carefully studied. Every version of DOS and Windows (except NT) comes with a DEFRAG program that will reorganize the directory, files, and free space to maximize performance. Effective use of RAM as a disk cache can address some of the I/O problems. Our QikFix program has a Memory Utilization program to help optimize the swap file and a Cache utility to optimize the disk cache. 12.6 Booting Process Whenever the computer is switched on the operating system gets loaded. The process of loading DOS is called as booting DOS. Once started up, the computer performs its self-test i.e. power-on hardware diagnostics. At this time, the computer examines most of its critical hardware components. The computer then displays the amount of memory present on the system. It checks the floppy drive (if available on the machine) to see if a floppy has been inserted. If there is no floppy, it checks the hard disk, loads IO.SYS and Ms DOS.SYS into the RAM and runs a file called CONFIG.SYS and then loads COMMAND.COM. These files are known as DOS system files. Thereafter, AUTOEXEC.BAT is executed.

Once the system boots up properly it displays the DOS prompt: C:\< (C:\ is known as C prompt. This is an indication for the user to know that the computer is ready for executing commands) In case a floppy is present while booting, it must be bootable, i.e., it should have the three main files COMMAND.COM, BIOS.SYS and MS DOS.SYS. The Booting process begins when the power supply is switched on. booting process performs following steps: 1. The power supply performs a self-test: 2. The microprocessor timer chip receives the "Power Good" signal: 3. The CPU starts executing the ROM BIOS code: 4. The BIOS searches for adapters (usually video adapters) that may need to load their own ROM BIOS routines: 5. The ROM BIOS checks to see if this is a 'cold boot' or a 'warm boot': 6. POST (Power-On Self-Test): The POST is a series of diagnostic tests that run automatically when you turn your computer on. 7. The BIOS locates and reads the configuration information stored in CMOS: 8. Shadow RAM: (Optional, you can turn it off/on using the CMOS settings) Shadow RAM is where a copy of BIOS routines from ROM is stored a special area of RAM, so that the BIOS routines can be accessed more quickly. 9. Loading the OS (Operating System): The BIOS will attempt booting using the boot sequence determined by the CMOS settings and examine the MBR (Master Boot Record) of the bootable disk. The MBR is the information in the first sector (512 bytes) of any hard disk or diskette that identifies how and where an operating system is located so that it can be loaded into the RAM (booted). The MBR is also sometimes called the "partition sector" or the "master partition table" because it includes a table that locates each partition that the hard disk has been formatted into. In addition to this table, the MBR also includes a program that reads the boot sector record of the partition containing the operating system to be booted into RAM. In turn, that record contains a program that loads the rest of the operating system into RAM.

Figure - MS-DOS Command Prompt 12.7 The Command Prompt When you first bring up the MS-DOS prompt, you'll see something that looks similar to the following: C:\WINDOWS< This is called the command prompt or DOS prompt. The flashing underscore next to the command prompt is called the cursor. The

cursor shows where the command you type will appear. If your command prompt looks like the sample command prompt above, skip to the following section, `Typing a Command.' If your command prompt does not look like the example, type the following at the command prompt, and then press ENTER: cd \ 12.8 Typing a Command This section explains how to type a command at the command prompt and demonstrates the `Bad command or file name' message.
➢
To type a command at the command prompt 1. Type the following at the command prompt (
you can type the command in either uppercase or lowercase letters): nul If you make a typing mistake, press the BACKSPACE key to erase the mistake, and then try again. 2. Press ENTER You must press ENTER after every command

you type. The following message appears on the screen: ➢ Bad command or file name The `Bad command or file name' message appears when you type something that MS-DOS does not recognize. Because nul is not a valid MS-DOS command, MS-DOS displays the `Bad command or file name' message. Note that you can also get this error if you try to run a program in DOS that doesn't exist or that DOS can't find. Now, type the following command at the command prompt: ver The following message (or something similiar) appears on the screen: Windows 98 [Version 4.10.1998] The ver command displays the operating system version you are running. Continue to the next section, where you will use the dir command to view the contents of a directory. 12.9 Drive Name Drive name: A drive is a device used for storing data for the further use and a computer can have more than one drive also for the ease of the user drive names or device name are given to a device or a drive. Drives are named by an alphabet example A drive, C drive etc. DESCRIPTION A : B : C : D : D : to Z : First removable disk drive Second removable disk drive First fixed disk drive (hard disk) Second fixed disk drive (hard disk) Additional disk drive 12.10 File Information is stored on a disk as a collection of character. Each complete collection of related characters is called a file. A disk can have many file that contain either instruction for the computer (program file) or data stored by the user (data file). A program file is loaded the internal memory of the computer whenever the user asks the DOS to load it. Once the program is loaded, the instructions are executed one by one by processor. A disk can contain both file and data files. 12.11 Directory The directory is a table of contents for a disk. It contain the name of files their size, and the dates they were last modified. In windows a directory is referred as Folder. A directory can contain a number of subfolders and file. 12.12 Files and File Name: Most Dos Commands perform some operation on, either the entire disk or on files stared on a disk data on a disk is organized into files. For identification, each one has a file name. On any one disk, each file name must be unique so that files don't get configured. The name of a file actually has tow parts called the file name proper and the file name extension. The file name must be at least one character and it can be as long as eight characters, but no longer. Alphabets digits and some special characters. &>+, %, (,), _, -, @, # (- are allowed in the file name. the extension to the file name is a short appendage added on after the file name. the two are separated by a period. The extension is three characters at mart, and is optional. The two are separated by a period. The extension is three characters at mart and it is optional. The intended purpose of an expansion is to indicate the category. 12.13 DOS Commands All the DOS COMMANDS fall into two categories. 1. Internal DOS Commands 2. External DOS Commands 12.13.1 Internal DOS Commands Internal DOS COMMANDS are those commands, which stay in the computer's memory. Until you switch it off or until, you load DOS again. Because they are loaded in memory, you do not need the DOS diskette in a drive. To use the internal commands.

12.13.2 External DOS Commands The rest of the commands on the DOS diskette are called external commands because they are not read into the computer's memory when you load DOS since the commands are not loaded in memory you need you DOS diskette in a drive to use the external commands. 12.13.3 Wildcard Characters A wildcard character is a keyboard character such as an asterisk (*) or a question mark (?) that you can use to represent one or more real characters when you are searching for files or folders. Wildcard characters are often used in place of one or more characters when you don't know what the real character is or you don't want to type the entire name. Asterisk (*) You can use the asterisk as a substitute for zero or more characters. If you're looking for a file that you know starts with gloss but you can't remember the rest of the file name, type the following: gloss* The Find dialog box will locate all files of any file type that begin with gloss including Glossary.txt, Glossary.doc, and Glossy.doc. To narrow the search to a specific type of file, type: gloss*.doc In this case, the Find dialog box will find all files that begin with gloss but have the file extension.doc, such as Glossary.doc and Glossy.doc. Question Mark (?) You can use the question mark as a substitute for a single character in a name.

For example, if you typed gloss?.doc, the Find dialog box would locate the file Glossy.doc or Gloss1.doc but not Glossary.doc. DIRECTORY COMMANDS DIR Command - FORMAT: DIR (>drive< (i>path name<) (>file name<) (/P) (/W) If you wish to see the contents of on directory, you may use the following commands It is used to list any or all the files and subdirectories in a directory. The drive, path, switches and wildcards can be used. It will display the file and other related information, if the file is present, otherwise it gives the message as file not found. DIR filename options : The options are preceded by a slash (/) which is also called a switch. /P - Pauses at the end of the screen and waits for a key to be pressed.

/S - Displays files in a specified directory and all subdirectories. /A - Displays all the files with set attributes like hidden files etc. C:\<DIR/P Wildcard characters, i.e., `*' (asterisk) and `?' (question mark) are also used. `*' can be used to replace characters anywhere from 1 to 8 (from that location to end of the extension) before the `.' and 1 to 3 characters after the `.' while `?' can be used to replace only one character. Figure - Output of DIR Command MD Command - FORMAT: MKDIR >DRIVE NAME> \ >NEW SUB-DIR NAME< FORMAT: MD >DRIVE NAME> \ >NEW SUB-DIR NAME< In this section, you will create two directories. Creating a directory is helpful if you want to organize related files into groups to make them easy to find. Before you begin this section, make sure the command prompt looks like the following: C:\< To create a directory, you will use the md command. The md command stands for `make directory'. ➢ To create and change to a directory named FRUIT 1. Type the following at the command prompt: md fruit You have now created a directory named FRUIT. You won't see the new FRUIT directory until you carry out the dir command in the next step. 2. To confirm that you successfullycreated the FRUIT directory, type the following at the

command prompt: dir or dir /p Look through the directory list. Anew entry somewhere in the list should look similar to the following: FRUIT >DIR< 09-07-00 11:59a fruit 3. To change to the new FRUIT directory, type the following at the command prompt: cd fruit The command prompt should now look like the following : C:\FRUIT< You will now create a directory within the FRUIT directory, named GRAPES. ➢ To create and work with a directory named GRAPES 1. Type the following at the command prompt: md grapes You will not see the new GRAPES directory until you carry out the dir command in the next step. 2. To confirm that you successfully created the GRAPES directory, type the following at the command prompt: dir A list similar to the following appears: Figure Output of MD Command Note that there are three entries in the FRUIT directory. One is the GRAPES directory that you just created. There are two other entries--one looks like a single period (.) and the other looks like a double period (..). These directory entries are important to MS-DOS, but you can ignore them (for now). They

appear in every directory and contain information relative to the directory structure. The GRAPES directory is a subdirectory of the FRUIT directory. A subdirectory is a directory within another directory. Subdirectories are useful if you want to provide further subcategories in the organization of your files and data. 3. To change to the GRAPES directory, type the following at the command prompt: cd grapes The command prompt should now look like the following: C:\FRUIT\GRAPES< 4. To switch back to the FRUIT directory, type the following: cd.. The command prompt should now look like the following: C:\FRUIT< When the cd command is followed by two periods (..), MS-DOS moves up one level in the directory structure. In this case, you moved up one level from the GRAPES directory to the FRUIT directory. This command gives a clue as to the meaning of the.. entry in a directory listing. The.. directory is a special name for the parent directory of the current default directory. That is why when you type cd.. you change directories to the parent directory. The single period (.) directory refers to the current directory. Try typing cd. and note what happens. The command cd. changed the current directory to the current directory! MD and MKDIR This lets you Make a Directory, hence the MD (MKDIR is a hold over from UNIX style OS's. If you switch between a UNIX OS and DOS, you may come to appreciate MKDIR). You type MD followed by a file name. You can nest (Make a directory within a directory) up to 16 directories on most DOS only systems (No Window). Some will let you nest up to 32. On window 95 machines you can get 3 lines of directories in a DOS box and 16 directories in MS-DOS mode. At this moment I really don't know where the limitation comes from, as it appears to be limited by the length of the command line. But I would think something other than the length of the command line would create the limitation (Like the FAT or something). The shorter the directory name the more you can nest. it is good practice to separate your files and programs so everything is organized.

CD Command Displays the name of the current directory or changes the current folder. FORMAT : chdir [/d] [drive:][path] [..] -or- FORMAT: cd [/d] [drive:][path] [..] Parameters none Used without parameters, chdir displays the names of the current drive and folder. Used with only a drive letter (for example, cd C:), chdir displays the current directory on the specified drive. [/d] Changes the current drive-in addition to changing the current directory for a drive. With command extensions enabled, the current directory name is converted to use the same case as the actual directory names on the disk. So, for example, cd C:\Temp sets the current directory to C:\Temp if that is its name on the disk. [drive:][path] Specifies the drive (if other than the current drive) and directory to which you want to change. [..] Specifies that you want to change to

the parent folder. Either of the following commands changes your current directory to the directory named Reports: chdir \reports cd \reports Suppose you have a directory named Specials containing a directory named Sponsors. To change your current directory to \Specials\Sponsors, type the following command: cd \specials\sponsors Or, if your current directory is \Specials, you can use the following command to change to the \Specials\Sponsors directory: cd sponsors To change from a directory back to its parent directory, type the following command: cd.. To display the name of the current directory, you can use chdir or cd without a parameter. For example, if your current directory is \Public\Jones on the disk in drive B, type chdir to see the following response: B:\Public\Jones !f you are working on drive D and you want to copy all files in the \Public\Jones and \Public\Lewis

directories on drive C to the root directory on drive D, type the following commands: chdir c:\public\jones copy c:*.* d:\ chdir c:\public\lewis copy c:*.* d:\ If, instead, you want to copy all files in the \Public\Jones and \Public\Lewis directories to your current location on drive D, type the following commands: chdir c:\public\jones copy c:*.* d: chdir c:\public\lewis copy c:*.* d: Notes - 1. Using spaces in paths with chdir With command extensions enabled, you can use the chdir command to change to a directory name that contains a space or spaces without surrounding the name with quotation marks. For example: cd \winnt\profiles\username\programs\start menu is the same as: cd "\winnt\profiles\username\programs\start menu" which is what you must type when extensions are disabled. 2. Changing to the root directory The root directory is the top of the directory hierarchy for a drive. To return to the root directory, type the following command: cd\ 3. Using the current directory from a different drive If you are working in the \Public\Jones directory on drive C and you change to drive D, you can copy files to and from the \Public\Jones directory by specifying only the drive letter C. 4. Changing the directory on another drive You can change the current directory on another drive by specifying the drive name on the command line when you use chdir. RD Command FORMAT: RD DIR >drive name< \>sub-dir-name<

If you no longer use a particular directory, you may want to delete it to simplify your directory structure. Deleting a directory is also useful if you type the wrong name when you are creating a directory and you want to delete the incorrect directory before creating a new one. In this section, you will delete the PARAS directory. Before you begin this section, make sure the command prompt looks like the following: C:\< To delete a directory, use the rd command. The rd command stands for `remove directory'. ➤ To delete the kamal directory 1. Type the following at the command prompt: rd kamal 2. To confirm that you successfully deleted the GRAPES directory, type the following at the command prompt: dir The PARAS directory should no longer appear in the directory list. Note that you cannot delete a directory if you are in it, or if it contains any files. Before you can delete a directory, you must make the directory that is one level higher the current directory (by typing cd..), plus delete any files in the directory if any exist. You will learn how to delete files later on in this tutorial. TREE Command - FORMAT : TREE [drive:][path] [/f] [/a] Tree Command graphically displays the directory

structure of a drive or Path in a hierarchical form. Parameters drive: Specifies the drive that contains the disk for which you want to display the directory structure. path Specifies the directory for which you want to display the directory structure. /f Displays the names of the files in each directory. /a Specifies that tree is to use text characters instead of graphic characters to show the lines linking subdirectories.

To display the names of all the subdirectories on the disk in your current drive, type: tree \ To display, one screen at a time, the files in all the directories on drive C, type: tree c:\ /f | more To print the same list that the previous example displayed, type: tree c:\ /f prn Figure - Output Of Tree Command Notes: The structure displayed by tree depends upon the parameters you specify on the command line. If you do not specify a drive or path, tree displays the tree structure beginning with the current directory of the current drive PATH Command FORMAT: PATH {>drive :<} {>path name<} : {>drive:} {>path name<}: This command set a command search path. If tells DOS which directories should be searched for external commands after searching the current directory. e.g. the command path\user\notice tells DOS to search the directory \USER\NOTICE For on external command after searching the current directory. You can specify several path names separated by : DOS Searches the path names in the order specified in the PATH command. e.g. PATH\USER\DASDIR:\USER\COBDIR:\USER\DBDIR. To deracinate the path command, you can specify PATH. This cause only the current directory to be searched for on external command. The following command specifies that DOS is to search three directories to find external

commands. The three paths for these directories are C:\User\Taxes, B:\User\Invest, and B:\Bin: path c:\user\taxes;b:\user\invest;b:\bin CD and CHDIR This command lets you change directories. Lets say you are at root (Just a "C:\<" prompt) and you want to get into a directory named WINDOWS. Type the following: C:\<cd windows You will get a prompt like this: C:\WINDOWS< Now, wasn't that fun?! But wait, there's more! Oh know!! You need to get back to your root directory, and quick! Type "CD.." and you will be magically transported back to root. Ok, go back into any directory again. Now type DIR to see what is in there. You will notice that there are two sub directories, one called "." and the other called "..". They represent where you were and where you are. The "." is the directory you are in. ".." represents the parent directory, or the directory this directory is in (Confused yet?! Hang on, it'll get better the more you use it!). When you type "CD.." you are telling DOS to go to the directory you where just in. If you type "CD." nothing will happen since you are in that directory already. Pop quiz: what if you are about 7 directories in and you need to get to root? Instead of typing CD.. seven times in a row, use "CD\". Another trick; lets say that you are in your "C:\DOS<" directory and you want to get into "C:\GAMES\NES<"? Use this: C:\DOS<cd \games\nes You will be sent directly into your NES directory within your GAMES directory. 12.14 File Management Commands COPY COMMAND - COPY Command is used to make copies of disk files. The diskette that contains the file you want to copy, is called the source diskette. The diskette that will contain the copy of the file when you are done is called the target diskette. There may following different uses of copy commands. This command copies files between disk drives and/or devices, either keeping the same file name or changing it. COPY can concatenate two or more files into another file or append one or more files to another file. Making duplicates on other disks: FORMAT: Copy A: >File name< B:>File Name< This command consists of three parts. First part is the command COPY, Second part

is the name of the file which is to be copied and the

third part is a parameter B: Which simple Says" COPY to Drive B"

If different file name is not specified at B: ;file will be copied with the same name as is at a : Make duplicates on the same disk. FORMAT: COPY A: >File Name< >File Name< Using the above command a file will by copied with different name on the same diskette. Examples The following command copies a file and ensures that an end-of-file character is at the end of the copied file: copy memo.doc letter.doc /a To copy a file named Robin.typ from the current drive and directory to an existing directory named Birds that is located on drive C, type the following command: copy robin.typ c:\birds If the Birds directory doesn't exist, DOS copies the file Robin.typ into a file named Birds that is located in the root directory on the disk in drive C. To copy several files into one file, list any number of files as source parameters on the copy command line. Separate file names with a plus sign (+) and specifya file name for the resulting combined file, as the following example shows: copy mar89.rpt + apr89.rpt + may89.rpt report This command combines the files named Mar89.rpt, Apr89.rpt, and May89.rpt from the current drive and directory and places them in a file named Report in the current directory on the current drive. When files are combined, the destination file is created with the current date and time. If you omit destination, DOS combines the files and stores them under the name of the first specified file. For example, if a file named Report already exists, you can use the following command to combine all four files in Report: copy report + mar89.rpt + apr89.rpt + may89.rpt You can also combine several files into one by using wildcard characters (* or ?), as the following example shows: copy *.txt combin.doc This command combines all files in the current directory on the current drive that have the extension.txt into one file named Combin.doc, also in the current directory on the current drive. If you want to combine several binary files into one by using wildcard characters, include the / b switch, as the following example shows: copy /b *.exe combin.exe This prevents DOS from treating CTRL+Z as an end-of-file character. Caution

➢ If you combine binary files, the resulting file might not be usable due to internal formatting. In the following example, copy combines each file that has a.txt extension with its corresponding.ref file. The result is a file with the same file name but with a.doc extension. Thus, copy combines File1.txt with File1.ref to form File1.doc. Then copy combines File2.txt with File2.ref to form File2.doc, and so on. copy *.txt + *.ref *.doc The following copy command combines first all files with the.txt extension, and then all files with the.ref extension into one file named Combin.doc: copy *.txt + *.ref combin.doc XCOPY COMMAND x-copy command Copies files and directories, including subdirectories. A DOS COPY command is used extremely to copy file form one disk or directory to another. DOS provides a second file-copy command, called X-COPY which allows you to- Selectively copy files based on the files creation and modification date. Selectively copy files based on the files archive attribute flag. FORMAT: xcopy source [destination] [/w] [/p] [/c] [/v] [/q] [/f] [/l] [/d[:date]] [/u] [/i] [/s [/e]] [/t] [/k] [/r] [/h] [/a|/m] [/n] [/exclude:filename] [/y | /-y] [/z] Parameters source Specifies the location and names of the files you want to copy. This parameter must include either a drive or a path. destination Specifies the destination of the files you want to copy. This parameter can include a drive letter and colon, a directory name, a file name, or a combination. /w Displays the following message and waits for your response before starting to copy files: Press any key to begin copying file(s) /p Prompts you to confirm whether you want to create each destination file. /c Ignores errors. /v

Verifies each file as it is written to the destination file to make sure that the destination files are identical to the source files. This switch is ignored because the functionality is inherent to the Windows 2000 operating system. The switch is accepted only for compatibility with previous versions of MSDOS. /q Suppresses display of xcopy messages. /f Displays source and destination file names while copying. /l Does not copy files, only displays (lists) files that would be copied. /d[:date] Copies only source

files changed on or after the specified date. If the date value is

missing, xcopy copies all source files that are newer than the time of existing destination files. This option allows you to update only files that have changed. If you specify a date, use a hyphen (-) as the separator rather than a forward slash (/) so that the date is not interpreted as another parameter. /u Copies (updates) only files from source that exist on destination. /i If source is a directory or contains wildcards, and destination does not exist, xcopy assumes destination specifies a directory name and creates a new directory then copies all specified files into the new directory. By default, xcopy will prompt you to specify whether destination is a file or directory. /s Copies directories and subdirectories, unless they are empty. If you omit this switch, xcopy works within a single directory. /e Copies all subdirectories, even if they are empty. Used with the /s and /t switches. /t Copies only subdirectory structure (tree), not files. To copy empty directories, you must include the /e switch. /k Copies files and retains the read-only attribute on destination files if present on the source files. By default, the read-only attribute is removed. /r Copies over read-only files.

/h Copies files with the hidden and system file attributes. The xcopy command does not copy hidden or system files by default. /a Copies only source files that have their archive file attributes set. This switch does not modify the archive file attribute of the source file. For information about how to set the archive file attribute, see the attrib command. /m Copies source files that have their archive file attributes set. Unlike the /a switch, the /m switch turns off archive file attributes in the files specified in source. For information about how to set the archive file attribute, click attrib in the Related Topics list[JG1]. /n Copies using NTFS short file or directory names. This switch is required when copying files or directories from an NTFS volume to a FAT volume or when the FAT file system naming convention (8.3) is required on the destination volume. The destination file system may be FAT or NTFS. /exclude:filename Excludes the files listed in the specified file from the copy operation. The exclusion file can have a list of exclusion patterns (one per line, no wildcard characters are supported). If any exclusion pattern in the file matches any part of the path of a subject file, that file is not copied. /y Suppresses prompting to confirm you want to overwrite an existing destination file.

The /y

switch may be preset in the COPYCMD environment variable. This may be overridden with /-y on the command line. The default is to prompt on overwrites unless the copy command is being executed from within a batch script.

To append files, specify a single file for destination, but multiple files for source (using wildcards or file1+file2+file3 format). /-

y Causes prompting to confirm you want to overwrite an existing destination file. /z Copies

over a network in restart able mode. If the connection is lost during the copy phase (for example, if the server going offline severs the connection), it will resume after the connection has been re-established. Using this command switch also displays the percentage of the copy operation completed for each file.

Examples To copy all the files and subdirectories (including any empty subdirectories) from the disk in drive Ato the disk in drive B, type: xcopy a: b: /s /e To include any system or hidden files in the previous example, add the/h switch: xcopy a: b: /s /e /h To update files in the \Reports directory with the files in the \Rawdata directory that have changed since December 29, 1993, type: xcopy \rawdata \reports /d:12-29-93 To update all the files that exist on \Reports in the previous example, regardless of date, type: xcopy \rawdata \reports /u To obtain only a list of the files that would be copied for the previous command, without copying the files, type: xcopy \rawdata \reports /d:12-29-93 /l < xcopy.out The file Xcopy.out lists every file that would be copied. To copy the \Customer directory and all subdirectories to the directory \\Public\Address on network drive H: and retain the read-only attribute, while being prompted when a new file will be created on H:, type: xcopy \customer h:\public\address /s /e /k /p To issue the previous command and ensure xcopy creates the \Address directory if it does not exist, without prompting, add the /i switch: xcopy \customer h:\public\address /s /e /k /p /i You can create a batch program to perform xcopy operations and use the batch if command to process the exit code in case an error occurs. For example, the following batch program uses replaceable parameters for the xcopy source and destination parameters: @echo off rem COPYIT.BAT transfers all files in all subdirectories of rem the source drive or directory (%1) to the destination rem drive or directory (%2) xcopy %1 %2 /s /e if errorlevel 4 goto lowmemory if errorlevel 2 goto abort if errorlevel 0 goto exit :lowmemory

echo Insufficient memory to copy files or echo invalid drive or command-line syntax. goto exit :abort echo You pressed CTRL+C to end the copy operation. goto exit :exit To use this batch program to copy all files in the C:\Prgmcode directory and its subdirectories to drive B, type: copyit c:\prgmcode b: The command interpreter substitutes C:\Prgmcode for %1 and B: for %2, then uses xcopy with the /e and /s switches. If xcopy encounters an error, the batch program reads the exit code and goes to the label indicated in the appropriate IF ERRORLEVEL statement. DOS displays the appropriate message and exits from the batch program. DEL Command - FORMAT: DEL>drive:< >file name< This command is used to remove specified files permanently from the disk. The file name can also be specified with the wild cards: del [drive:][path] filename […] [/p] [/f] [/s] [/q] [/a[:attributes]] erase [drive:][path] filename […] [/p] [/f] [/s] [/q] [/a[:attributes]] Parameters [drive:][path] filename Specifies the location and name of the file or set of files you want to delete. Multiple file names can be used. File names can be separated by spaces, commas, or semicolons. /p Prompts you for confirmation before deleting the specified file. /f Forces deletion of read-only files. /s Deletes specified files from the current directory and all subdirectories. Displays the names of the files as they are being deleted. /q Quiet mode; does not prompt for delete confirmation.

/a Deletes files based on specified attributes. attributes Can be any of the following file attributes: Attribute Description r Read-only a Archive s System h Hidden - Prefix meaning "not" Note ➢ If you disable command extensions, the /s switch is reversed so that instead of displaying the names of files that are being deleted, it displays the names of any files that were not found. Notes 1. Using the /p switch If you use the /p switch, del displays the name of a file and prompts you with a message in the following format: filename, Delete (Y/N)? Press Y to confirm the deletion, N to cancel the deletion and display the next file name (if you specified a group of files), or CTRL+C to stop the del command. 2. Deleting more than one file at a time You can delete all the files in a folder by typing the del command followed by [drive:]path. For example: del \work deletes all files in the \Work folder. You can also use wildcards (* and ?) to delete more than one file at a time. However, you should use wildcards cautiously with the del command to avoid deleting files unintentionally. Suppose you type the following command: del *.* The del command displays the following prompt:

All files in directory will be deleted! Are you sure (Y/N)?

Press Y and then ENTER to delete all files in the current folder, or press N and then ENTER to cancel the deletion. Before you use wildcards with the del command to delete a group of files, you can use the same

wildcards with the dir command to see a list of the names of all the files included in the group. =< Warning ➢ After you delete a file from your disk, you may not be able to retrieve it. If command extensions are enabled, the /s option displays as its output the names of the files that are deleted. Ordinarily, it shows the names of files it could not find. Examples To delete all the files in a folder named Test on drive C, you can use either of the following commands: del c:\test del c:\test*.* RENAME COMMAND - FORMAT : rename [drive:][path] filename1 filename2 FORMAT: ren [drive:][path] filename1 filename2 This command is used to Change the name of a file or a set of files. You can rename all files matching the specified file name. You cannot use the rename command to rename files across drives or to move files to a different directory location. Parameters [drive:][path] filename1 Specifies the location and name of the file or set of files you want to rename. filename2 Specifies the new name for the file. If you use wildcards (* and ?), filename2 specifies the new names for the files. You cannot specify a new drive or path when renaming files. Notes - 1. Using wildcards with rename You can use wildcards (* and ?) in either file name parameter. If you use wildcards in filename2, the characters represented by the wildcards will be identical to the corresponding characters in filename1. 2. Rename will not work if filename2 already exists If, for filename2, you specify a file name that already exists, rename displays the following message: Duplicate file name or file not found

Examples Suppose you want to change the extensions of all the file names in the current directory that have the extension.txt; for example, you want to change the.txt extensions to.doc extensions. To make this change, type: ren *.txt *.doc To rename a file or directory named Chap10 to Part10, type: ren chap10 part10 ATTRIB COMMAND - FORMAT: [drive:] [path] ATTRIB [+A] [-A] [+R] [-R] file spec. This command is an external command is used to changed the attribute of a file that is, you can use ATTRIB to make a file "read only" which prevents the file from being written over (or changed), also you can set the archive but, which is set tone when a file is crated or changed. This but is used by the BACK UP and X COPY commands to determine if now or changed files exist. These two commands provide a time saving option that updates only those files that require updating FORMAT: [drive:] [path] ATTRIB [+A] [-A] [+R] [-R] file spec. ATTRIB uses +R to set and remove the read only attributes. The +A and - A options let you set and clear the archive attribute of a file. First you can use ATTRIB to determine the attribute status of a file by typing ATTRIB the file name, and pressing return. Notes 1. Groups of files You can use wildcard characters (? and *) with the filename parameter to display or change the attributes for a group of files. If a file has the System or Hidden attribute set, you must clear that attribute before you can change any other attributes for that file. Archive attributes The Archive attribute (+a) is used to mark files that have changed since they were previously backed up. The xcopy command uses these Archive attributes. For information about Archive attributes for xcopy, see Related Topics. Examples To display the attributes of a file named News86 located on the current drive, type the following command:

attrib news86 To assign the read-only attribute to the file named Report.txt, type the following command: attrib +r report.txt To remove the read-only attribute from files in the \Public\Jones directory on a disk in drive B and from files in any subdirectories of \Public\Jones, type the following command: attrib -r b:\public\jones*.* /s As a final example, suppose you want to give an associate a disk containing all files in the default directory on a disk in drive A except files with the.bak extension. Because you can use xcopy to copy only those files marked with the archive attribute, you need to set the Archive attribute for those files you want to copy. To do this, you use the following two commands to set the Archive attribute for all files on drive Aand then to clear the attribute for those files with the.bak extension: attrib +a a:*.* attrib -a a:*.bak Next, use the xcopy command to copy the files

from the disk in drive A to the disk in drive

B. The /a switch in the following command causes xcopy to copy only those files marked with the Archive attribute: xcopy a: b: /a If you want xcopy to clear each file's Archive attribute after it copies the file, use the /m switch instead of /a, as in the following example: xcopy a: b: /m BACKUP COMMAND - FORMAT: [drive:] [Path] BACK UP SOURCE: [Files pec] Target: [A] {/M} [/S] [/F] [/ D:mm_dd_yy] [/T hh: mm: ss] [/L: Log file] The BACK UP Command is used with fixed disk systems. BACK UP is on external command to copy files automatically from a fixed disk to happy disk. BACK UP let you copy selected files, all files in a specified directory path, or every file on the disk. It also lets you copy only those files that have been changed or saved after a specifies date. It the files being copied exceed the capacity of the target floppy disk. BACK UP prompts you change disks and then to continue the process. Files can be divided between floppy disks. The backup command automatically tags each back up disk in the sequence used. Later, you can used the RESTORE Command to copy the files back to the fixed disk. Having each disk numbered helps you restore the fixed disk in the right order. e.g. BACK UP C: \ PATH A:

This form copies all files in the specified directory path to the disk in drive A. There is a no. of parameters available with the BACK UP Command. These are: Is causes all files in sub directories of the specified path to be copied CLS COMMAND The CIS commands stand for clear the screen. It wipes the screen of every thing on it and finishes up by displaying the DOS prompt in the top left car nor. FORMAT: CLS SYS COMMAND FORMAT.: C:\<sys a: This command is used to make a boot disk. It copies IO.SYS, MSDOS.SYS, and COMMAND.COM to your boot disk. With older versions of DOS the diskette needed to be completely empty before you used the SYS command. That was because the system files (MSDOS.SYS and IO.SYS with Microsoft's DOS; IBMBIO.COM and IBMDOS.COM for IBM's DOS) had to be the first 2 files on the disk. Since DOS 5 that isn't necessary anymore. You can also make a boot disk by using the "/S" option with where a: is the drive you are "sys'ing" One VERY important note, for SYS to work it must know where to find the system files. By default it looks on the current drive you are on (If you're on the C: drive, it looks to C:\ for the system files.) If the drive you are currently in does not have the system files, SYS may give you "Invalid function" or "Can not find system files". If your current drive doesn't have these files, you can tell SYS where to find them like so: C:\<sys c:\ a: where "c:\" is the location of your system files, and "a:" is the drive you are wanting to make a boot disk in Rather than trying to remember the second example, just be sure you always use SYS from your C: drive (As long as you boot off your C: drive!) One more note, you can't see the system files with just DIR, you need to use DIR /A/P or they will be invisible. (That's for windoze 95/98 users, for users of DOS 5 through 6.22, use DIR, -yes include the comma! If you have a DOS below 5, yer on yer own!)

12.15 General Commands TYPE COMMAND FORMAT: Type >drive :<>path< >file name< Displays the contents of a text file. Use the type command to view a text file without modifying it. type [drive:][path] filename Parameter [drive:][path] filename Specifies the location and name of the file or files that you want to view. Separate multiple file names with spaces. Notes 1. Displaying binary files If you display a binary file or a file created by a program, you may see strange characters on the screen, including formfeed characters and escape-sequence symbols. These characters represent control codes used in the binary file. In general, avoid using the type command to display binary files. Examples To display the contents of a file named Holiday.mar, type: type holiday.mar To display the contents of a lengthy file one screen at a time, type: type holiday.mar | more DATE COMMAND FORMAT: date [mm-dd-yy] DATE command is used to make it possible to change the current date. Correct data is recorded is the directory when you create or change a file. This command displays the system date and asks to enter a new date and then changes the system date if the user does not enters the new date the date does not changes If the user enters an invalid date the DOS returns an error invalid date asks to enter the date again. FORMAT: DATE Example To set the date to January 12, 1994, type: date 01/12/1994 To display the current system date, type:

date /t TIME COMMAND- TIME commands permits you to enter or change the time known to the system. When ever you create or add to a file, the time is recorded in the directory : This command displays the current system time and asks the user to enter the new time if is to be modified Syntax A:< Time Example à A:.< Time Current time 9:35:20.16a Enter new time : This command asks the user to enter new time and changes the system time if the user does not enters the new time the time does not changes A:<Time Current time 9:35:20.16a Enter new time : FORMAT: DATE PROMPT COMMAND FORMAT: [Prompt---text] The PROMPT COMMAND provides you with a mechanism to set a new DOS-Prompt Where prompt text can be text to be displayed as the new system prompt and can contains special meta- strings in the form of $ c, where meaning are defined in Table. META-STRINGS MEANINGS $ The "$" Character The Time T The Time D The Date V Dos Version No. N Dos Drive Letter G The "<" Character I The ">" Character B The " : " Character

9 The "=" Character H A Back Space The Previous E The Escape Character - The C R L F (Carriage Return Line Feed) New Line On Sequence, Which Cause You To The Beginning To Go To Cause You To Go Of "A" The PROMPT command within any parameters cause returned to the normal DOS PROMPT DIRECTORY ENTRIES The no. of entries that the root directory of a disk can contain depends on the type of disk. The root directory of a single sided 160/180KB diskette can hole 84 entries. Which the root directory of a double sided 320/360KB and 720 KB diskette can hold 112 entries. The root directory of a height capacity. 1.2 MB diskette can hold 224 entries. There can be 512 entries in the root directory of a fixed disk. Unlike the root directory, sub directory subdirectories can contain any no. of entries limited only by the amount of available space on the disk. The correct directory is the one you are currently working in or the one you were working in one a different drive DOS Remembers which directory was current on each of your drives, even though you may not be presently averring any of those drives. When DOS starts the root directory is the current directory. Examples The following example sets the command prompt to display the current drive and path followed by the greater-than symbol (<): prompt pg The following command displays a two-line prompt in which the current time appears on the first line and the current date appears on the second line: prompt time is: t_date is: $d 12.16 Disk Management Commands FORMAT COMMAND FORMAT: FORMAT(>drive:<) (/S) (IV) (14) Format actually does two important things with a diskette: it puts identification mark on the disk for storage/retrieval that make it possible for DOS to work with the diskette and it checks for any defects in the diskette. It also prepares the disk to accept DOS files by initializing the root directory $ file allocation table.

The DOS FORMAT command prepares a disk for use. It divides a disk into two areas. System area DOS stores information which is of use to itself User area Stores programs or data files System area consists of the boot area (which is the first sector on the disk), the file allocation table and the root directory. Syntax : FORMAT drive :[/V[:label]][/Q][/V][/S] Where, drive :specifies the drive containing the disk you want to format Switches : /V[:labes] : Specifies the volume label. /Q : Performs a quick format. /V : Perform an unconditional format. /

F: size : Specifies the size of the floppy disk to format. /

B : Allocates space on the formatted disk for system file. /S : Copies system file to the formatted disk. /

T :track : Specifies the number of tracks per disk side. /I : Formats a single side of a floppy disk. /4 : Formats 5.25 inch 360K floppy disk in a high density. /8 : Formats 8 sectors per track. /

C : Tests sectors that are currently marked bad. /N: Sector : Specifies the number of sectors per track. Examples C:<FORMAT B: Formats the disk in drive B: C:<FORMAT B:/S After formatting the System file will be transferred to drive B so that with the disk booting can be done. Notes 1. Using Format

The format command creates a new root directory and file system for the disk. It can also check for bad areas on the disk, and it can delete all data on the disk.

For DOS

to be able to use a new disk, you must first use this command to format the disk. 2.

Typing a volume label After formatting a floppy disk, format displays the following message: Volume label (11 characters, ENTER for none)? The volume label can be a maximum of 11 characters (including spaces). If you do not want your disk to have a volume label, just press ENTER. For information about

volume labels, click label in the Related Topics list. 3. Formatting a hard disk When you use the format command to format a hard disk, Windows DOS displays a message of the following form before attempting to format the hard disk: WARNING, ALL DATA ON nonremovable DISK DRIVE x: WILL BE LOST! Proceed with Format (Y/N)?_ To format the hard disk, press Y; if you do not want to format the disk, press N. You must have Administrator rights to format a hard disk. 4. Unit size FAT file systems restrict the number of clusters to no more than 65526. FAT32 file systems restrict the number of clusters to between 65527 and 268435445.

NTFS compression is not supported for allocation unit sizes above 4096. 5.

Format messages When formatting is complete, DOS displays messages showing the total disk space, any space marked as defective, and the space available for your files. 6. Quick formatting You can speed up the formatting process by using the /q switch. Use this switch only if there are no bad sectors on your hard disk. 7. Using format with a reassigned drive or a network drive You should not use the format command on a drive prepared by using the subst command. You cannot format disks over a network. 8. Format exit codes The following list shows each exit code and a brief description of its meaning: 0 The format operation was successful. 1 Incorrect parameters were supplied. 4 A fatal error occurred (any error other than 0, 1, or 5). 5 The user pressed N in response to the prompt "Proceed with Format (Y/N)?" to stop the process. You can check these exit codes by using the error level condition with the if batch command.

Examples - To format a new floppy disk in drive A, using the default size, type the following command: format a: To perform a quick format operation on a previously formatted disk in drive A, type the following command: format a: /q To format a floppy disk in drive A and assign it the volume label "DATA," type the following command: format a: /v:DATA CHKDSK COMMAMD - FORMAT: chkdsk [drive:][[path] filename] [/f] [/v] [/r] [/l[:size]] [/x] Creates and displays a status report for a disk, based on the file system used. Chkdsk also lists and corrects errors on the disk. If chkdsk cannot lock the drive it will offer to check it the next time the computer restarts. Issuing the chkdsk command on a fixed disk requires you be a member of the Administrators group. Parameters none Used without parameters, chkdsk displays the status of the disk in the current drive. drive: Specifies the drive that contains the disk that you want chkdsk to check. [path] filename Specifies the location and name of a file or set of files that you want chkdsk to check for fragmentation. You can use wildcard characters (* and ?) to specify multiple files. /f Fixes errors on the disk. The disk must be locked. If chkdsk cannot lock the drive it will offer to check it the next time the computer restarts. /v Displays the name of each file in every directory as the disk is checked. /r Locates bad sectors and recovers readable information. The disk must be locked. /l[:size] NTFS only. Changes the log file size to the size you enter. Displays the current size if you don't enter a new one.

/x NTFS only.

Forces the volume to dismount first, if necessary. All open handles to the volume

are then invalid. This switch also includes the functionality of the /f switch. /i NTFS only. Performs a less vigorous check of index entries, reducing the amount of time needed to run chkdsk. /c NTFS only. Skips the checking of cycles within the folder structure, reducing the amount of time needed to run chkdsk. Examples If you want to check the disk in drive D and have DOS fix any errors encountered, type the following command: chkdsk d: /f Chkdsk pauses and displays messages if it encounters errors. Chkdsk finishes by displaying a report showing the status of the disk. Also, no files can be opened on the drive specified. On a FAT disk, to check all files in the current directory for noncontiguous blocks, type chkdsk *.* Chkdsk displays a status report, then lists the files matching the file specification that have noncontiguous blocks. DISKCOPY COMMAND - FORMAT: DiskCopy >source drive:< >destination drives:< Copies the contents of the floppy disk in the source drive to a formatted or unformatted floppy disk in the destination drive. Parameters drive1 Specifies the drive containing the source disk. drive2 Specifies the drive containing the destination disk. /v Verifies that the information is copied correctly. Use of this switch slows the copying process. For information about copying one or more files, click copy in the Related Topics list. For information about copying directories and subdirectories, click xcopy in the Related

Topics list. Example - To create an exact copy of the disk in drive B on the disk in drive A, type diskcopy b: a: LABEL COMMAND - FORMAT: LABEL>DRIVE:< (Volume Label) This command allows you to create, change or delete a volume label on disk. Volume tables are used to identify disk. They can be up to 11 characters. To change the existing volume label, type the new volume label and press enter. Notes - 1. Label command messages If you do not specify a label when you use the label command, DOS displays a message in the following format: Volume in driveAis xxxxxxxxxxx Volume Serial Number is xxxx-xxxx Volume label (11 characters, ENTER for none)? The "Volume Serial Number" part of the message is not displayed if the disk has no serial number. You can type the volume label you want or press ENTER to delete the current label. If a disk has a label and you press ENTER for none, DOS prompts you with the following message: Delete current volume label (Y/N)? Press Y to delete the label; press N to keep the label. 2. Limitations on volume label names A volume label can contain as many as 32 characters for NTFS volumes and as many as 11 characters for FAT volumes and can include spaces but no tabs. FAT volume labels cannot contain any of the following characters: * ? / \ |., ; : + = [ ] > < " This limitation does not apply to NTFS volumes. DOS displays FAT volume labels in uppercase letters. If you enter a volume label in lowercase letters, the label command converts the letters to uppercase. NTFS volume labels retain and display the case used when the label was created.

Example To label a disk in drive Athat contains sales information for July, you might type the following: label a:sales-july VOL COMMAND -FORMAT: VOL >

drive< Displays the disk volume label and serial number, if they exist.

Aserial number is displayed for a disk formatted with MS-DOS version 4.0 or later. Parameter drive: Specifies the drive that contains the disk for which you want to display the

volume label and serial number. 12.17 DOS Utility Commands MOVE COMMAND - Syntax: MOVE >SOURCE FILENAME < >TARGET FILENAME< This command moves one or more files to another directory. It is also used for renaming a directory. It will move the source file under a new name, i.e., target file name C:\<MOVE ABC XYZ >RETURN< Moves ABC to XYZ in the root directory itself. After execution of this command the screen will look like: C:\MOVE ABC XYZ >RETURN< ABC =< XYZ OR MOVE PATH\SOURCEFILENAME(s) PATH It will move the source file from its original path to the new directory under a new name, if a new name is mentioned. Example To move all files with the.xls extension from the \Data directory to the \Second_Q\Reports directory, type move \data*.xls \second_q\reports\

DELTREE COMMANDS This nifty command was unveiled upon the arrival of DOS 5. Unlike it's counter part RD it can remove directories even when they still have files and subdirectories in them. Some consider this a very dangerous command, I just say make backups :-) It accepts a sole switch, "/Y". Normally DELTREE will ask you for conformation before it does its magic. The "/Y" switch prevents this. SETVER COMMAND Sets the MS-DOS version number that the MS-DOS subsystem reports to a program. setver [drive:path] [filename n.nn] setver [drive:path] [filename [/delete [/quiet]] To display the current version table, use the following syntax: setver [drive:path]

Paramet ers

none Used without parameters, the setver command displays the current version table. [drive:path] Specifies the location of the Setver.exe file. filename Specifies the name of the program file (.exe or.com) that you want to add to the version table. You cannot use a wildcard (* or ?). n.nn Specifies the MS-DOS version (for example, 3.3 or 4.01) that the MS-DOS subsystem reports to the specified program file. /delete Deletes the version table entry for the specified program file. You can abbreviate this switch as /d. /quiet Hides the message typically displayed during deletion of an entry from the version table. Examples - Suppose you have a program file named Myprog.exe that runs with MS-DOS version 3.30. To run Myprog.exe, you must first use the following setver command to create an entry in the version table that will cause Myprog.exe to interpret the MS-DOS subsystem as version 3.30:

setver myprog.exe 3.30 To delete the Myprog.exe entry from the version table (without otherwise affecting the Myprog.exe file), type: setver myprog.exe /delete To list the contents of the version table on drive C, type: setver c: 12.18 Using Copycon to Build a File - FORMAT : COPY CON >FILE NAME< Copycon command is used to create a file in the current directory. Exmple: C:\copy con myfile Note : to save the file myfile use ^z figure : creating a file using copy con command Summary ➢ The head of disk is a small coil and reads or writes on the position of the disk rotating below it, therefore, the data is stored in concentric set of rings. These are called tracks. The width of a track is equal to the width of the head. ➢ Dos have three system file (1) COMMAND.COM (2) IO.SYS (3) MSDOS.SYS. ➢ In MS Dos

The user configuration files are CONFIG.SYS and AUTOEXEC.BAT. The reader is assumed to be familiar with these files. ➢ The file allocation table is a table of entries describing how each cluster is allocated on the disk. Each cluster on disk has a entry. The FAT is simply a table of numbers. The entry in FAT indicates whether the corresponding cluster is in use or not. It holds a pointer to the next

cluster. ➢ Whenever the computer is switched on the operating system gets loaded. ➢ The process of loading DOS is called as booting DOS. Once started up, the computer performs its self-test i.e. power-on hardware diagnostics. ➢ The directory is a table of contents for a disk. It contain the name of files their size, and the dates they were last modified. In windows a directory is referred as Folder. A directory can contain a number of subfolders and file. Questions 1. What are the internal and External commands in MS-DOS. 2. What do you understand by the term "Booting the System". 3. what is config.sys file in MS-DOS. 4. What is the use of Autoexec.bat file. 5. What are the wild card character in MS-DOS. 6. Write the command to print a file in MS-DOS. 7. How will you rename a DOS file. 8. Describe in detail the usage of COPY command. Give the example of combining two or more files into another file using COPY command. 9. Describe with example the purpose and usage of DELTREE command. 10. Describe the DIR command. What are the switches used with DIR command. 11. Describe the purpose of FDISK command. 12. what is the use of LABEL command. 13. what is FAT. 14. What is the use of PATH command. 15. What is the use of FORMAT command.

Unit 13: Windows Operating System 13.0 Objectives 13.1 Introduction 13.2 Features of MS-WINDOWS 2000 13.3 Windows 2000 Professional at a Glance 13.4 Accessibility for special needs 13.5 Hardware Requirements of WINDOWS 2000 13.6 Desktop Environment 13.7 Taskbar 13.8 Start Menu 13.9 My Computer 13.10 Recycle Bin 13.11 Control Panel 13.12 Windows Explorer 13.13 Windows Accessories 13.14 Managing Hardware and Software and its Installation 13.15 Add/Remove Applications (Softwares) 13.16 Add New Hardware 13.17 Printers 13.18 System Tools (Disk Drive Utilities) 13.19 Disk Defragmenter 13.20 Check for Disk Errors (SCAN DISK) 13.21 Backup Your Data 13.22 Clipboard Viewer 13.23 System Information 13.24 NTFS 13.25 Network Neighbourhood (My Network Places) 13.26 Sharing Information between Programs 13.0 Objectives After going through this unit to will be able to:

➢ Describe Features of Windows, taskbar, toolbar, desktop, recyclebin, control panel etc. ➢ Describe File folders management, windows explorer, installing and running programms, configuring mouse and keyboard, managing software and hardware. ➢ Describe setting printer and fonts, backup, sound configuration network management ➢ Describe cilpboard, system information, NTFS, sharing information between programms. 13.1 Introduction MS-Windows 2000 is the most popular GUI for personal computers. Windows provides an environment that enhances DOS in many ways. Microsoft Windows 2000 Professional, the premier desktop operating system for businesses and organizations. Windows 2000 provides faster performance, greater reliability, improved security, and a more manageable desktop. With its broader support for hardware and applications, Windows 2000 is the best platform for running the latest software and hardware. No matter where you are working, your computer will be easier to use and to manage, because Microsoft Windows 2000 Professional is more compatible and more powerful than any workstation you've used before. 1. Easier to use With Windows 2000 Professional, you have faster access to information, and you are able to accomplish tasks more quickly and easily. Windows 2000 Professional makes it easier to: ● Work with files. ● Find information. ● Personalize your computing environment. ● Work on the Web. ● Work remotely. 2. Easier to manage You and your network administrators can work more efficently now, because many of the most common computer-management tasks are automated and streamlined with Windows 2000 Professional. With Windows 2000, your workstation will be easier to: ● Set up. ● Administer.

● Support. 3. More compatible Windows 2000 Professional offers increased compatibility with different types of networks and with a wide array of legacy hardware and software. Windows 2000 also provides: ● Improved driver support. ● Increased support for new-generation hardware and multimedia technologies. ● Integration of the new Euro currency symbol. 4. More powerful For all your computing needs, Windows 2000 Professional provides: ● Industrial-strength reliability. ● The highest level of security. ● Powerful performance. 13.2 FEATURES OF MS-WINDOWS 2000 The major benefits of Windows are: 1. Common Look and Feel: All Windows applications have the same basic look and feel. Once you know one or two Windows applications. It is easy to learn another one. 2. Device Independence: Windows presents a device-independent interface to applications. Unlike most of today's DOS applications, a Windows application is not bound to the underlying hardware such as mouse, keyboard or display. Windows shields the applications from this responsibility. The application deals with the Windows API to manipulate any underlying devices. 3. Memory Management: Windows also provides memory management to break the 640K limitation of MS-DOS. An application has the ability to use the extended memory, share data segments with other applications and swap unwanted segments to disk. 4. Support for existing DOS applications: Windows allow most standard DOS applications to run under it directly. Any application that does not control the PC's hardware, use the PC BIOS or MS-DOS software interrupts, can run in its own window. 5. Data Sharing: Windows allows data transfer between application Clipboard. Any type of data can be transferred from one window with the Clipboard. The Dynamic Data Exchange (DDE) protocol defines how two applications can share information. Information such as bitmap, metafile, character strings and other data formats can be shared.

6. True 32-bit Operating System: The computer stores and processes data in bits, which is the smallest unit for measuring information. OS can be classified on the basis of the number of bits it can process at a time. Windows NT is a 32-bit OS, which implies that it can send or receive 32 bits of information at a time and can internally process 32 bits of data. 7. Portability: The characteristics of the microprocessor affect the OS design. An OS which is designed with a particular processor in mind can take advantage of the specific features of that processor, but will not be able to run on computers which use any other processor. The ability of software to run on different processors is referred to as portability. Windows NT is one such portable OS, which can run on both the older Intel 80486 processors as well as on the latest Pentium and MIPS R 400 processors. 8. Scalability: Windows NT is a multi-processing OS, which means that it can use more than one microprocessor for its internal processing. If such a processor is available in the system unit. In fact, Windows NT can run on computers with as many as 32 processors. The architecture of Windows NT is designed to take maximum advantage of the additional processors and ensure that none of the processors is idle. 9. Multitasking Features: Windows NT is a multitasking OS, which means that it can perform more than one task at the same time. For example, you can writer a sales report in MS-WORD, and can refer to the latest sales figures by having the data file open. While you are writing this report, your system could also be printing a document in the background. Multitasking is implemented by using the principle of time-sharing, where each application is allowed to run only for a fixed duration of time. After that, the microprocessor puts that task on a queue and performs another task. Thus, it manages to perform more than one task at a time. This feature is

also known as preemptive multitasking. 10. Multiple User Support: Windows NT is also a multi-user OS, which means that is supports more than one user. Several users can access the same files; share the same resources, like printers or scanners. However at the same time, Windows NT ensures that no user can use another user's files without authorization or interfere with the tasks being executed by another user. 11. Multi-Threading: The multiple tasking and multiple user support provided by windows NT is implemented by the principle of multiple threading. A thread is defined as a single sequential flow of control. A program is normally broken into small chunks called tasks, and tasks are further broken into smaller chunks called threads. Every process has a thread that determines the system resources it can access and the operation that it can perform on

it. Windows NT allows a process to have more that one thread associated with it. For example, you can edit a file, print in the background and another user may also be working on the same file simultaneously. All these processes seem to be running simultaneously because each of these processes is different thread. This feature makes multi-tasking and multi-user support possible. 12. Built-in Networking: Windows NT can be used as a server, with several workstations, known as nodes, connected to it. The nodes can communicate, access resources and data on the server as well as on other nodes. The networking support also enables Windows NT to make use of microprocessors of other computers on a network. This ensures maximum utilization of system resources. This feature is known as distributed processing. 13. Graphical User Interface: Windows NT 4.0 has a user-friendly graphical interface. Users can now perform complex tasks by selecting options from drop-down menus and click on icons to execute programs. 14. Extensibility The extensibility of window NT provides the flexibility of adding faster workstations and servers to the network. 15. Localization Window NT is being offered in many countries around the world in local languages. Thus users located in different countries can interact with the system in their native language. Windows NT supports date, time and currency formats of different countries. RESOURCES Resources are used to manage windows and user-defined objects, MS-Windows provides nine kinds of resources to application developers. These resources are : icons, cursors, menus dialog boxes, fonts, bitmaps, char strings, user-defined resources, and keyboard accelerators. 1. Icons and cursors: Windows defines a few types of icons and cursors. An icon or a cursor is essentially a bit-mapped region that is used to represent and symbolize a window or cursor. A developer can also define an original icon or cursor using the ICON EDIT utility. 2. Menus : Each window can have its own menu bar. A menu item can be a character string or a bitmap. Each item of a menu bar in turn can have a pop-up menu presenting a list of options. Currently, Windows does not support nesting of pop-up menus within other pop-up menus. (Windows 3.0 provides this functionality). But a pop-up menu can invoke a dialog box. 3. Dialog boxes : These provide another mechanism besides pop-up menu and menu bars to obtain information from the end-user. Dialog boxes are much more flexible than menu bars or pop-up menus. Dialog boxes usually contain a group of child windows such - as buttons, scroll bars, and editable fields. Just like windows, dialog boxes have a function that is used to process messages received from the user upon selection of options 4. Fonts : Windows provides a few families of fonts with different sizes and shapes : modem, roman, Swiss, Helvetica, and script. Application processors and desktop publishing can define additional fonts as needed. 5. Bitmaps : They are used to represent icons, cursors, or draw picture on the screen. Both mono and color bitmaps can be defined. 6. Character Strings : Character strings are handled as resources mainly to provide a manageable solution to internationalization of a window application. 7. User-Defined Resources : These can be used for any purpose and support any user- defined data type. Any arbitrary data can be managed as a user-defined resource. 13.3 Windows 2000 Professional at a Glance Windows 2000 is the most manageable, reliable, and secure version of Windows ever. Improved hardware support and increased software compatibility make using Windows 2000 even easier. And enhancements to networking, printing, and storage help you find resources more quickly and work more efficiently. Easiest Windows Yet Desktop and window enhancements make Windows 2000 easy to use. You can customize menus and toolbars to tailor Windows 2000 to your needs. Microsoft Windows Explorer displays more information about files and folders, including thumbnail views and encryption information. Using the enhanced search capabilities, you can browse and see search results in the same window. In My Documents and My Pictures folders you have a convenient place to back up and store all of your work. And Microsoft Internet Explorer gives you the best Internet experience and latest Web browsing features. For example, the AutoComplete feature finishes typing Internet addresses and previously-viewed files for you. The AutoDetect feature detects and configures a proxy server for your connection. Windows 2000 includes hundreds of new printer, modem, and other hardware drivers, making hardware

installation and setup easy and efficient. New wizards make configuring hardware simple. The Add/Remove Hardware wizard automatically detects and configures new devices. The Add Printer wizard helps you change printer settings from within applications. And Windows 2000 includes improved printing support-Internet printing, Image Color Management 2.0, and new font technologies. Windows 2000 Professional is the best operating system for mobile users. The Network Connection wizard simplifies setting up network and dial-up connections. You can take network-based files and folders offline, making them available when you aren't connected to the network. Likewise, you can save Web pages for offline browsing. And Synchronization Manager ensures that offline and network versions of a file are up-to-date. Building on the Traditional Strengths of Windows NT Windows 2000 significantly expands the existing capabilities of Windows NT, providing a more stable and responsive computing experience. The new Encrypting File System ensures that information is completely private and secure, even for users who share a computer. You can now more easily create a virtual private network to securely connect computers over the Internet. Windows 2000 supports the following scalable security technologies: Kerberos V5 protocol, smart card, public key cryptography, and Internet Protocol security (IPSec). You need only log on once to use all Windows 2000 Server network resources. With the latest installation technology, you can install, customize, and remove programs more reliably. Windows Installer helps prevent system file conflicts, repairs incomplete installations, updates out-of-date components, and removes all traces of a program, including registry entries. For more efficient disk space allocation, Windows 2000 now supports the file allocation table (FAT) 32 file system. The NTFS file system includes many performance enhancements and new features such as per- user disk quotas and file encryption. Information travels faster with improved TCP/IP stack performance, reducing network traffic. Files built with dynamic Hypertext Markup Language (DHTML) load more quickly. You can accommodate the latest transmission technologies, including cable modems, through support for asynchronous transfer mode technology. Best of Windows 98 Windows 2000 contains key Windows 98 features, such as Advanced Configuration and Power Interface power management, the DirectX technologies, and Web integration. It's as easy to install and use state-of-the-art hardware with Windows 2000 as it was with Windows 98 because Windows

2000 also supports Plug and Play, IEEE 1394, DVD, and the universal serial bus (USB) standards. And as in Windows 98, in Windows 2000 you can use multiple monitors with a single computer to dramatically increase the size of your workspace. Windows 2000 is compatible with more applications than ever before, continuing to support software for previous versions of Windows-as well as older hardware. To ensure that you're using the highest quality hardware drivers, the drivers are digitally signed by Microsoft after testing. Also, Windows 2000 is ideally suited to support the demands of new programs, including distributed programs written using Windows Distributed internet Applications (Windows DNA) architecture. Lower Total Cost of Ownership Windows 2000 can reduce the total cost of ownership-the costs involved in administering personal computer networks-compared with that of other versions of Windows. Remote installation, a disk imaging tool, support for Windows 95 and Windows 98 upgrades, and the unattended Setup tools help administrators easily deploy Windows 2000, saving time and expense. With Windows Script Host, you can also automate tasks with language-independent scripts. And Microsoft Management Console (MMC) consolidates several administrative programs into a single interface you can use to manage and support local and remote computers. The breakthrough IntelliMirror technology uses the power of Windows 2000 Server to make it easier for administrators to manage desktops remotely-each user's documents, system files, and administrative settings are stored on a server. Administrators are able to quickly duplicate a Windows 2000 installation from one computer to another. Whether administering small businesses or large corporations, you can manage desktops centrally using Active Directory service and its Group Policy support in Windows 2000. Administrators can deploy software applications using group policies. Support for Windows Management Instrumentation, a program management tool, reduces the maintenance cost of event reporting and makes managing device drivers easier. And you can keep Windows 2000 current by using the Windows Update Web site, an extension of Windows 2000. From this Web site, you can download system enhancements, such as drivers and service packs. Multilanguage technology lets you view, print, and share documents from any localized language supported in Windows 2000. Asingle version of Windows 2000 Professional can be deployed globally while still allowing users to access information in their preferred language.

13.4 Accessibility for special needs You can adjust the appearance and behavior of Windows 2000 to enhance accessibility for some vision-impaired, hearing-impaired, and mobility-impaired users without requiring additional software or hardware. Windows 2000 includes the following programs to enhance accessibility: ➢ Magnifier enlarges a portion of the screen for easier viewing. ➢ Narrator uses text-to-speech technology to read the contents of the screen aloud. This is useful for people who are blind or who have low vision. ➢ On-Screen Keyboard provides users with limited mobility the ability to type on-screen using a pointing device. ➢ Utility Manager enables users with administrator-level access to check an Accessibility program's status, start or stop an Accessibility program, and designate to have the program start when Windows 2000 starts. Using Accessibility Options in Control Panel, you can easily set up how you want to use your keyboard, display, and mouse functions. The accessibility tools available in Accessibility Options in Control Panel perform various functions: ➢ StickyKeys enables simultaneous keystrokes while pressing one key at a time. ➢ FilterKeys adjusts the response of your keyboard. ➢ ToggleKeys emits sounds when certain locking keys are pressed. ➢ SoundSentry provides visual warnings for system sounds. ➢ ShowSounds instructs programs to display captions for program speech and sounds. ➢ High Contrast improves screen contrast with alternative colors and font sizes. ➢ MouseKeys enables the keyboard to perform mouse functions. ➢ SerialKeys allows the use of alternative input devices instead of a keyboard and mouse. You can use the Accessibility wizard to help you set up the options and programs for your individual needs. Please see Related Topics for information on using these Accessibility aids. A wide variety of hardware and software products are available to make personal computers easier to use for people with disabilities. Among the different types of products available for MS-DOS and the Microsoft Windows operating systems are: ➢ Programs that enlarge or alter the color of information on the screen for people with visual impairments. ➢ Programs that describe information on the screen in Braille or synthesized speech for

people who are blind or have difficulty reading. ➢ Hardware and software utilities that modify the behavior of the mouse and keyboard. ➢ Programs that enable the user to type using a mouse or his or her voice. ➢ Word or phrase prediction software, that allows users to type more quickly and with fewer keystrokes. ➢ Alternate input devices, such as single switch or puff-and-sip devices, for people who cannot use a mouse or a keyboard. The Accessibility tools that ship with Windows 2000 are intended to provide a minimum level of functionality for users with special needs. Most users with disabilities will need utility programs with higher functionality for daily use. 13.5 Hardware Requirement of Windows 2000 Before you install Windows 2000, make sure your computer meets the following minimum hardware requirements: ➢ 133 MHz Pentium or higher microprocessor (or equivalent). Windows 2000 Professional supports up to two processors on a single computer. ➢ 64 megabytes (MB) of RAM recommended minimum. ➢ 32 MB of RAM is the minimum supported. 4 gigabytes (GB) of RAM is the maximum. ➢ A 2 GB hard disk with 650 MB of free space. ➢ If you're installing over a network, more free hard disk space is required. ➢ VGAor higher resolution monitor. ➢ Keyboard. ➢ Microsoft Mouse or compatible pointing device (optional). For CD-ROM installation: ➢ A CD-ROM or DVD drive. ➢ High-density 3.5-inch disk drive, unless your CD-ROM drive is bootable and supports starting the Setup program from a CD. For network installation: ➢ Windows 2000-compatible network adapter card and related cable (see the Hardware Compatibility List, Hcl.txt, in the Support folder on the Windows 2000 Professional CD). ➢ Access to the network share that contains the Setup files.

13.6 Desktop Environment When you start Windows 2000 for the first time, you see icons displayed on the screen. The Windows 2000 screen is a Desktop. The desktop is the first view of the Windows 2000 operating system. The desktop contains icons, shortcuts and the taskbar. The user can navigate the user interface using the taskbar or the start menu. For starting WINDOWS 2000, simply switch-on the computer and you will come directly to WINDOWS 2000 screen (as shown in the figure given below). This screen is called the desktop (metaphor), the settings of which can be changed as the user likes To adjust settings such as desktop color and background, use the right mouse button to click anywhere on the desktop, and then click Properties. The icons on the left side represent some of the accessories and tools. They provide the short and easier access to the features that we will discuss in the later sessions. The bar that is at the bottom of the screen is called the taskbar. Figure: Desktop of windows 2000 13.7 Taskbar The Taskbar is a bar that appears at the bottom of the windows 2000 desktop by default. The start button is at the left end of the taskbar. All the applications that are running appear as minimized buttons on the taskbar. The current window is displayed as a pressed button: all other applications are displayed as raised buttons. Clicking on a button for an inactive application activates the application and displays its window as the current window. The taskbar thus makes it easier to switch between applications. figure: task bar

13.8 Start Menu The start menu is a menu located at the left end of the taskbar. Start menu is integral to the clean look of the Windows NT desktop because it helps you to minimize the clutter on your desktop. Clicking on the start button opens a popup menu that makes help, the run command, a list of programs and a list of recently accessed documents available to you, to execute with a single click. From the start menu, you can customize the look and feel of windows, find files and folders and shut down your computer. For some items, submenu opens to the side of the main item to display the list of choices Figure : Start menu Launching an Application To start an application, select the Programs option and click on the program application you want to execute. The Programs option displays a submenu with a list of sub options as displayed in Figure The Programs option usually displays Accessories, Startup, Command Prompt, and Windows Explorer as a part of its submenu. The Accessories sub option includes Games, Multimedia, system Tools, Calculator, Calendar, Paint, Hyper Terminal and WordPad. This option is used to launch various applications. Alternatively, application can also be launched through the Run option of the Start menu. The user needs to specify the name of the executable file and the application is launched. Figure 4 shows the Run window.

Figure: Run Menu DOS AS AN APPLICATION Windows NT is a new operating system that is being used by a large number of computer users all over the world. Many companies however still use MS-DOS based applications. Therefore, supports for these applications were a key design goal. Windows NT provides compatibility with MSDOS based applications. In Windows NT, DOS is an application by name CMD. EXE. This application displays a simulated screen of the DOS command prompt in the form of a window. To start command prompt application in the Windows NT server ➢ Select the programs option from the start menu ➢ Click command prompt in the programs sub-menu The command prompt, which appears her, is C :\<. (Refer FIGURE 5) If drive C is currently active, and you want to access the files on a diskette in drive A, you can change to drive Aby specifying A: at the cursor position. Figure: Command Prompt Similarly, to change to drive C type C: and press >enter<. 13.9 My Computer The My Computer window displays icons representing the various derives like A: C: etc and resource icons like printers, dial-up networking and the control panel. The various drive icons let you access the contents of each of the drives. The printers icon is a folder represent in the collection of printers installed in the system. It also lets the user install a new printer. Every window thus three buttons on the right-upper cornet that can be used to minimize, maximize and close the window. In a maximized window, the maximize button is replaced by the restore button. FIGURE 2 displays the My Computer window. Use My Computer when you want to view the contents of a single folder or disk. A disk is a device, such as a hard disk or floppy disk, on which you can store files. When you double-click My Computer on your desktop, icons representing the available drives appear in a new window. When you double-click a drive icon, a window displays the folders contained on that drive. You can then double-click an icon to open the folder or file. Some of the following icons may appear in the My Computer window. Double- click To View the contents of the floppy disk, which is usually designated as A. View the contents of the hard disk, which is usually designated as C. View the contents of a network drive, if your computer is connected to one. View the contents of a compact disc in the CD-ROM drive, if your computer has one. View features you can use to modify your computer settings. Set up a printer and view information about available printers and print job status. Schedule or view tasks for computer maintenance. View the contents of a folder. View files and folders on a Web server, if your computer is connected to one. Figure: My Computer

13.10 Recycle Bin Windows stores deleted files in the Recycle Bin. Which is located on the desktop? You can use the recycle bin to retrieve files that you have accidentally deleted, or empty the recycle bin to create more disk space. When you delete a file or folder, it is removed form the folder where it used to be. It moves the file from the folder to a storage area, known as the recycle bin. You see the recycle bin as an icon on your desktop (REFER FIGURE) Figure: Recycle Bin Icon Windows NT moves all deleted objects - files folders, shortcuts, and printers - into the recycle bin as a precautionary measure. In case a file is deleted accidentally, you can double-click on the recycle bin icon and move the file (s) back to their original location by selecting the restore option from the file menu. However, if the files have to be deleted, they should be deleted from the Recycle Bin. Figure 21 shows the Recycle Bin window and the options that the file menu provides for managing the files in the Recycle Bin. Figure: Recycle Bin Folder 13.11 Control Panel Control Panel is a convenient way to customize your computer. From Control Panel you can add and remove programs, fonts, and hardware; control how hardware, such as your mouse and modem, behaves; and select folder, display, and sound options.

Figure: Control Panel To open a Control Panel item 1. Click the Start button, point to Settings, and then click Control Panel. 2. Double-click the icon of the feature you want to open. Choosing Folder Options Use Folder Options in Control Panel to change how your desktop appears and how you browse through your files and folders. You can choose Web-style options, keep familiar options from earlier versions of Windows, or choose a combination of both. You can use Folder Options to choose whether: ● The Active Desktop is on or off. ● Web-like content appears in folders. ● You browse folder content in multiple windows or a single window. ● You double-click or single-click to open files. To change your folder options You can also double-click My Computer, and then on the Tools menu, click Folder Options. 1. Click the Start button, point to Settings, and then click Control Panel. Control Panel appears. 2. Double-click the Folder Options icon. The Folder Options dialog box appears.

3. On the General tab, click the folder options you want, and then click OK. 12.12 Window Explorer Windows Explorer displays the hierarchical structure of files, folders, and drives on your computer. It also shows any network drives which have been mapped to drive letters on your computer. You can also view My Network Places, which lists other computers that are connected to your local area network (LAN). Using Windows Explorer, you can copy, move, rename, and search for files and folders. For example, you can open a folder that contains a file you want to copy or move, and then drag it to another folder or even another drive. There are other places in Windows 2000 where you can view and work with files and folders. My Documents is a convenient place to store documents, graphics, or other files you want to access quickly. When you delete files or folders from your hard disk, Windows 2000 places them in the Recycle Bin, where you can retrieve them. Files or folders deleted from a floppy disk or a network drive are permanently deleted and are not sent to the Recycle Bin. For information about My Documents or Recycle Bin, see Related Topics. You can also navigate in Windows 2000 by using the taskbar (including the Start button), My Computer, Windows Explorer, or My Network Places. In some cases, one way is more efficient than another. For example, it's faster to start programs from the Start button than from Windows Explorer, but Windows Explorer gives you a more complete picture of all of the files on your computer. In other cases, the choice is personal preference. When browsing through folders (containers for documents and programs), for instance, you can use either My Computer or Windows Explorer-it's a matter of how you prefer to display the contents of your computer. The following sections familiarize you with some of your choices. Figure: Wiondow Explorer

To create a new folder -

1. Open Windows Explorer. 2.

Click the drive or folder in which you want to create

a new folder. 3.

On the File menu, point to New, and then click Folder. 4. Type

a name for the new folder, and then press ENTER. Note ➢ To open Windows Explorer, click Start, point to Programs, point to Accessories, and then click Windows Explorer. ➢ You can also create a new folder by right-clicking a blank area in the right pane of Windows Explorer or on the desktop, pointing to New, and then clicking Folder. To select multiple files and folders - 1. Open Windows Explorer. 2. Click the drive or folder you want to work with, and then do one of the following: ➢ To select consecutive files or folders, click the first item, press and hold down SHIFT, and then click the last item. ➢ To select files or folders that are not consecutive, press and hold down CTRL, and then click each item. ➢ To select all the files and folders in the window, click the Edit menu, and then click Select All. To copy or move

a file or folder - 1. Open Windows Explorer. 2. Click the drive or folder you want to work with. 3. Click the file or folder you want to

copy or move. 4. On the Edit menu, click Copy.

Or, click Cut to move the item. 5. Open the folder or disk where you want to copy or move the item. 6. On the Edit menu, click Paste.

Notes ➢ To select consecutive files or folders to copy or move, click the first item, press and hold down SHIFT, and then click the last item. ➢ To select files or folders that are not consecutive, press and hold down CTRL, and then click each item. To move files by dragging 1. Open Windows Explorer.

2. Find

the file or folder you want to move. 3. Make sure the destination for the file or folder you want to

move is visible. 4. Drag the file or folder to the destination. Notes ➢ If you drag using the right mouse button, a menu appears with the commands Move Here, Copy Here, Create Shortcut(s) Here, and Cancel. Click the command you want. ➢ To copy the item instead of moving it, press and hold down CTRL while dragging. ➢ If you drag an item to another disk, it is copied, not moved. To move the item, press and hold down SHIFT while dragging.

To delete a file or folder - 1. Open

Windows Explorer. 2. Click

the file or folder you want to delete. 3. On the File

menu,

click Delete. Notes ➢ Deleted files remain in the Recycle Bin until you empty it unless you press and hold down SHIFT while dragging an item to the Recycle Bin. Then the item is deleted from your computer without being stored in the Recycle Bin. ➢ If you want to retrieve a file you have deleted, double-click Recycle Bin on the desktop, right-click the file you want to retrieve, and then click Restore. ➢ You can also delete files or folders by right-clicking the file or folder and then clicking Delete. To create a shortcut in a folder - 1. Open Windows Explorer. 2. In the Folders list, click the folder in which you want to create the shortcut. 3.

On the File menu, point to New, and then click

Shortcut. 4. Follow the instructions on your screen. Notes ➢ To open Windows Explorer, click Start, point to Programs, point to Accessories, and then click Windows Explorer. ➢ Folders can also be opened by double-clicking the folder in the details pane on the right. ➢ A shortcut is a quick way to start a frequently used program or open a file or folder without having to go to its installed location. For information about creating shortcuts

on the desktop or Start menu, see Related Topics. To change the name of

a file or folder 1. Open Windows Explorer. 2.

Click the file or folder you want to rename. 3. On the File

menu, click Rename. 4. Type the new name,

and then press ENTER.

Notes ➢ You do not need to open the file or folder to rename it. ➢ A file name can contain up to 215 characters, including spaces. However, it is not recommended that you create file names with 215 characters. Most programs cannot interpret extremely long file names. File names cannot contain the following characters: \ / : * ? " > < | ➢ You can also rename a file or folder by right-clicking it and then clicking Rename. ➢ The name of system folders such as Documents and Settings, Winnt, or System32 cannot be changed because they are required for Windows to run properly. To search for a file or folder 1. Click Start, point to Search, and then click For Files or Folders. 2. In Search for files or folders named, type all or part of the file name or folder you want to find. 3. To search for files containing specific text, in Containing text, type the text you want to find. 4.

In Look in, click the drive, folder, or network you want

to search. 5. To specify additional search criteria, click Search Options, and then click one or more of the following options to narrow your search: ➢ Select Date

to look for files that were created or modified on or between

specific dates. ➢ Select Type to look for files of a specific type, such as a text or WordPad document. ➢ Select Size to look for files of a specific size. ➢ Select Advanced Options to specify additional search criteria. 6. Click Search Now. Notes ➢ If you get too many results, try entering additional search criteria to make your search

more specific. ➢ To clear the search criteria fields and begin a new search, click New. ➢ Under Advanced Options, Search slow files refers to files that reside on removable storage media, such as optical or tape backup. These files may need to be copied to a faster storage medium before the contents can be searched. To change file or folder properties 1. Open Windows Explorer. 2. Click the file or folder whose properties you want to change. 3. On the File menu, click Properties. Notes ➢ You can also right-click a file or folder and then click Properties. ➢ To change properties for archiving, indexing, compression, or encryption on NTFS drives, right-click the file or folder, click Properties, click Advanced, and then select the options you want to change. 13.13 Windows 2000 Accessories In the following session, we will discuss the various accessories in more detail. RUN YOUR PROGRAMS To run the accessory programs, follow the instructions: 1. Click the Start button. 2. Then click on Programs, and then click on Accessories. 3. The cascading menu appears. Click on the program you want to run Figure: Window Accessories To run other programs, that have been created by the users : 1. Click on Run... option from the Start menu. 2. Run dialog box is opened. In the Open text box, enter the path name of the program you want to run. 3. If you do not remember the name of the file, click on Browse... button. 4. In the Browse window, look in text box lists the available drives, folders, and files. Click on the down-arrow. 5. To see inside a folder, click it. The box below shows the folders and files in the selected location. 6. Double-click a folder or file to open it. 7. To open the folder one level higher, click on the toolbar. 8. Once the file is located, click on Open and then click on OK from the Run window. General Use The usual accessory programs that are used are easy and useful to work with. They are provided for performing certain day-to-day activities of the users. Calendar Windows Calendar is used to view the current month's calendar that shows the day and date, and also the current day showing the timings to keep track of your appointments. To display either of the mentioned views, click on the View option, and select whichever you require. Figure: Calender Above given are the samples of two views. The left window displays the day schedule to track your appointments, whereas, the right window displays the month calendar. Calculator You can use Calculator in standard view to do simple calculations or use it in scientific view to do advanced scientific and statistical calculations. You have been using this program in earlier versions of Windows also.

Figure: Calculator CHARACTER MAP Character Map works only with Windows-based programs. This is used to insert a special character into a document. To insert a special character in the document, follow the given steps: 1. Select Character Map from the cascading menu of Accessories. 2. Click the Font box, and then click a font. 3. To magnify a character in Character Map, click it and hold down the mouse button. Double- click each character you want. The character(s) will appear in the Characters To Copy box. 4. Click Copy. 5. In your document, click where you want the character(s) to appear, click the Edit menu, and then click Paste. 6. Select the characters, and then change them to the same font you used in Character Map. Figure: Character Map

Paint Paint is a bitmap-painting program with a full set of painting tools and a wide range of colors. You can use it to create, edit, and view pictures. The pictures created by Paint can be pasted into another document you have created or use it as your desktop background. You can even use Paint to view and edit scanned photos To start Paint, select Paint from the cascading menu of Accessories. The Paint window consists of a workspace, or canvas, where you paint pictures. To the left of the canvas is the Toolbox, which contains a set of painting tools. The color palette is at the bottom of the canvas. The selection box is on the left side, where you select the width of lines or pen tips to use for the Brush, Line, Eraser, and other tools. To paint, select a tool, a color, and a line width, and then start painting on the canvas. Wordpad WordPad is a text editor for short documents. Though, this accessory does not have the advanced features of full-blown word processors like MS-Word. You can format documents in WordPad with various fonts and paragraph styles. To start WordPad, click on Start menu, then on Accessories, and then on WordPad. Figure: Wordpad Once WordPad is running, you can do the following: 1. Specify the initial page layout for a document, such as paper size and margin settings. 2. Type, edit, and delete text, as well as copy or move text from one place to another. You can also copy or move text between two different applications. 3. Change the font, style, and size of characters, and change the alignment and indents of paragraphs. You can also set special tabs and create bulleted or numbered lists. 4. Search for previously typed text or replace text with new text.

5. Create compound documents that contain pictures, spreadsheet data, charts, sound, and video created in other applications. NOTEPAD You can use Notepad to create or edit text files that do not require formatting and are smaller than 64K. Notepad opens and saves text in ASCII (text-only) format only. To create or edit files those require formatting or are larger than 64K, use WordPad. Notepad loads very quickly and is easy to learn and use. To start Notepad, click on Notepad from the cascading menu of Accessories. Figure: NOTEPAD Now you can type text in the workspace and use standard editing techniques to insert, delete, and copy or move text. Entertainment Multimedia in Windows 2000 The previous versions of Windows and DOS were not designed to support multimedia, therefore, the multimedia presentations were either not running or running very slowly. With WINDOWS 2000, things are improved as it supports multimedia at the base level, as discussed below: ➢ Multimedia presentations require more than one media, and WINDOWS 2000 is a 32-bit, multitasking and multithreading operating system. This assures playing more than one media smoothly, and more responsive to user interaction. ➢ During installation procedure of WINDOWS 2000, it automatically detects and configures itself for multimedia devices. ➢ WINDOWS 2000 allows you to create compound documents that include sound, video, graphics, charts, pictures, and other elements from various applications. ➢ A new CD-ROM file system in WINDOWS 2000 improves CD-ROM performance.

➢ An AutoPlay feature makes it easy to set up and play back programs on CD-ROM. ➢ The video standard in WINDOWS 2000 is widely supported in the computer industry. ➢ WINDOWS 2000 supports larger video windows and higher frame rates. ➢ WINDOWS 2000 supports Sony VISCA interface which means that you and your applications can control VCRs and laser disc players. ➢ New graphic programming interface improves the game performance in WINDOWS 2000. ➢ WINDOWS 2000 supports a variety of industry-standard sound and video compression codecs (coders / decoders). Codecs reduce the size of multimedia files. ➢ All the multimedia controls can be changed or edited from the Multimedia Properties dialog box. To open this box, open the Control Panel and double-click the Multimedia utility. Figure: MULTIMEDIA PROPERTIES Multimedia Tools WINDOWS 2000 offers accessories for editing and listening to sound, playing movies, and playing music CDs. To access these accessories, follow the given steps: 1. Open the Start menu. 2. Select Programs from the menu. 3. From the cascading menu, click on Accessories. 4. Then click on Multimedia from the cascading menu of Accessories.

The options of the Multimedia menu are - CD player, Media Player, Sound Recorder, and Volume Control. Each of these is discussed in the subsections. CD PLAYER You can use CD Player to play audio compact discs from a CD-ROM drive connected to your computer. To use CD Player, you also need a sound card. Fig : CD PLAYER Before playing a CD make sure it is in the CD-ROM drive, and then click the start button as shown in Figure. Your CD continues to play even if you minimize CD Player. MEDIA PLAYER You can use Media Player to play audio, video, or animation files, and to control the settings for multimedia hardware devices. To use Media Player also, you should have a sound card. Figure: MEDIA PLAYER Sound Recorder The Sound Recorder is a handy but somewhat limited digital recorder. Using Sound Recorder you can

record, play, and edit sound files. To use Sound Recorder, you must have a sound card and speakers installed on your computer. If you want to record live sound, you also need a microphone. This multimedia utility can be used to create recorded messages and include them in files or include them in electronic mail messages that you send to other users. Figure: Sound Recorder To record a new sound, follow the steps given: 1. Attach a microphone to the soundboard or a patch cable from another sound source, such as a CD audio player or stereo. 2. To customize recording options, choose Properties from the File menu. 3. Choose recording format in the Format Conversion field. 4. Click the Convert now button that opens the Sound Selection dialog box. 5. In the Name field, specify the quality of sound from the pull-down list - CD Quality, Radio Quality, or Telephone Quality. For every option of the Name field that you select, note the contents of the Format and Attributes fields, which show the recording options and disk space requirements in seconds. VOLUME CONTROL If you have a sound card, you can use Volume Control to control the volume and speaker balance when you play audio files. To adjust the volume for multimedia devices, open the Multimedia Properties dialog box. In the Playback or Recording area, drag the Volume slider.

Figure: VOLUME CONTROL 13.14 Managing Hardware and Software And Its Installation System Settings The System can be personalized by making interesting and useful changes to WINDOWS 2000 settings. For example, you can : ● Adjust the double-click speed for your mouse. ● Change number, currency, time, and date settings. ● Change printer settings. ● Change settings for network service. ● Change background of your desktop. ● Change the capacity of the Recycle Bin. ● Change the number of colors your monitor displays. ● Have your monitor automatically turn off. ● Change the screen resolution. ● Change display fonts. ● Protect your screen by setting-up a screen saver. ● Configure multimedia devices. ● Enable multiple users to personalize settings. ● View or change resource settings for a hardware device. CONTROL PANEL When you click on the option Control Panel from the cascading menu of Settings, a window is opened that contains the icon of the utilities for changing hardware configurations or customizing the WINDOWS 2000 graphical interface as shown below

: Figure CONTROL PANEL You can also access the utilities of Control Panel, if you click on My Computer and then on Control Panel icon. 13.15 ADD/REMOVE APPLICATIONS (SOFTWARES) The WINDOWS 2000 components, accessories, or other applications can be added or installed, and removed by using the Add/Remove Programs utility from the Control Panel window. The advantage of using this utility is that WINDOWS 2000 maintains control over the installation process by making proper entries into the registry. ● To install an application, click on Install... button. While installing the application, the wizard helps you to complete the process step-by-step. ● To add a program, select it from the displayed list of programs and then click on Add/Remove button. ● To remove the application, again select it from the displayed list of applications and programs and click on Add/Remove button. 13.16 ADD NEW HARDWARE To set up new hardware ● Click on the Add New Hardware icon (as shown) to start Add New Hardware wizard. Figure: Simply, follow the instructions on your screen. It is recommended that you let Windows detect your new hardware. Make sure you have connected your hardware or installed its components in your computer before running the wizard.

13.17 Printers WINDOWS 2000 has significant advances in printing features. These are: ● It supports bi-directional communication with printers so that printers can provide WINDOWS 2000 with information about their current state and attributes. ● The WINDOWS 2000 printing system also returns control to users more quickly after a job is sent to the printer due to a new print spooler. ● Mobile and remote users can defer their printing job until later if they are not connected to a printer. Print jobs are stored in a local queue and automatically sent to the printer when they reconnect with the printer or reconnect to a network that has printers. Figure: PRINTER CONTROLS To change printer settings 1. Select the object Printers from the Control Panel window. 2. From the Printers window, click the icon for the printer you are using. 3. On the File menu, click Properties. 4. The settings you can change depend on the type of printer you have. Click the different tabs to see all of the options you can set. Changing the printer properties will change them for all documents you print on this printer. To change these settings for one document, use the Page Setup or Print Setup command on the File menu in your program.

To set up a new printer 1. Select Add Printer icon from the Printers window. 2. Add Printer Wizard is opened, and follow the instructions on the screen. 3. If you want to print a test page, first make sure your printer is on and ready to print. If you want to use a shared network printer, you can set it up quickly by browsing for it in Network Neighborhood, clicking the printer's icon, and then clicking Install on the File menu. 13.18 System Tools (Disk Drive Utilities) WINDOWS 2000 includes utilities to format diskettes, Check diskettes and hard drives for errors, defragments disks, and do other types of maintenance. Some of these utilities are discussed below. To access these utilities 1. Open the Start menu and select Programs. 2. Choose Accessories and then System Tools. The cascading menu that appears includes the various Disk Drive utilities. Figure: SYSTEM TOOLS 13.19 Disk Defragmenter Disk Defragmenter is used to rearrange files and unused space on the hard disk so that programs run faster. To speed up your hard disk by using Disk Defragmenter, follow the instructions given below:

1. From the cascading menu of System Tools, select Disk Defragmenter. Select Drive dialog box is opened to select the drive you want to defragment. Figure: DEFRAGMENTER 2. Click the drive you want to defragment, and then click OK. 3. If you want to change the settings that Disk Defragmenter uses, click Advanced.... 4. Click Start. While Windows defragments the selected disk, the computer can safely carry out other tasks. However, the computer will operate more slowly. To temporarily stop Disk Defragmenter so you can run other programs at full speed, click Pause. 13.20 Check For Disk Errors (Scan Disk) You can use Scandisk to check your hard disk for logical and physical errors, and then repair the damaged areas. To check your disk's surface, files, and folders for errors 1. Select the option Scandisk from the cascading menu of System Tools. 2. Click the drive you want to check. 3. Click Thorough. If you want to change the settings Scandisk uses when checking the disk's surface, click Options. If you want to change the settings Scandisk uses when checking files and folders, click Advanced. 4. Click Start.

Figure: SCAN DISK If you want to specify how Scandisk repairs any errors it finds, make sure that the Automatically Fix Errors box is not checked. The Standard type of test is only to check errors in files and folders but does not check the disk surface. DRIVE SPACE You can use DriveSpace to compress both hard and floppy disks to create more free space for files. You can also use DriveSpace to configure disk drives that you have already compressed by using DoubleSpace or DriveSpace. To determine how much space is available on a disk 1. Double-click the My Computer icon, and then click the disk you want to check. 2. On the File menu, click Properties. A pie chart shows how much free and used space is on the disk. To create more disk space by using DriveSpace 1. In the Drives On This Computer list, click the drive you want to compress. 2. On the Drive menu, click Compress. 3. Click Start. 4. If you have not backed up your files, click Back Up Files, and then follow the instructions on your screen. When you are done, proceed to step 6.

5. Click Compress Now. 6. If Windows prompts you to restart your computer, click Yes. 7. If you want to free up more disk space after your computer restarts, start the Disk Space Troubleshooter again. 13.21 Backup your Data You can use Backup to back up files on your hard disk. You can back up files to floppy disks, a tape drive, or another computer on your network. When you have made a backup file, you can restore it if your original files are damaged or lost. There are two ways you can backup your files : ● You can backup files on your hard disk to a floppy disk or other media by using simple copy techniques. ● The Microsoft Backup utility can also be used, which compresses files and puts them in an archive. These compressed files can be copied to a floppy drive, tape drive or other backup device. To start the Backup utility 1. Click Start and select Programs. 2. Select Accessories from the cascading menu, and then click on System Tools. 3. Now click on Backup. If this utility is not present in the cascading menu of System Tools, then probably this utility was not installed when you set up the system. Figure : BACK UP & RECOVERY To install the Backup utility, follow the instructions given below : 1. Start the Add/Remove Programs utility from the Control Panel window.

2. Click the Windows Setup tab and click the Disk Tools option in the window. 3. Then click OK to install the Backup utility, and then follow the instructions on the screen. While restoring backup files, again the Microsoft Backup utility is used. Restoring the files may pose a problem if you need to restore them on a system that does not have this utility or is not running WINDOWS 2000. In that case, other backup method may be used. 13.22 Clipboard Viewer When you cut or copy information from a program, it is moved to the Clipboard and remains there until you clear the Clipboard or until you cut or copy another piece of information. The Clipboard window in ClipBook Viewer shows the contents of the Clipboard. You can paste the information from the Clipboard into any document as often as you like. However, the information is only stored on the Clipboard temporarily. Using ClipBook Viewer, you can cut or copy information from another program and store it in a page that you can name, save, use again, and share with others. You can share ClipBook pages with others who have ClipBook Viewer installed on their computers, and they can share their ClipBook pages with you. You can also save the contents of the Clipboard to an individual Clipboard file instead of a ClipBook page. These files generally have the extension.clp and cannot be shared with others. The Clipboard can store text in different formats, such as the Text character set (used by most Windows 2000 and Windows NT-based programs), the OEM Text format (used by MS-DOS-based programs), and the Unicode character set (a superset of all of the major scripts used in the world). The View menu gives you the ability to see your data in various formats. In Windows 2000, Clipboard Viewer has been renamed Clipbook Viewer. ClipBook Viewer shows you the information you have copied to the Clipboard. You can store information permanently in your Local ClipBook and share it with other users. The Local ClipBook opens when you start ClipBook Viewer. For information about using ClipBook Viewer, see Related Topics or click the Help menu in ClipBook Viewer. Note ➢ To open ClipBook Viewer, click Start, click Run, and then type clipbrd. System Information In Windows 2000, you can use System Information in Computer Management to collect and display

your system configuration data. Support technicians require specific information about your computer so that they can resolve your system problem. You can use System Information to quickly find the data they need. For information about using System Information, see Related Topics or click the Action menu in Computer Management, and then click Help. 1. Open Computer Management (Local). 2. In the console tree, double-click System Tools, and then click System Information. Note ➢ To open Computer Management, click Start, point to Settings, and then click Control Panel. Double-click Administrative Tools, and then double-click Computer Management. 13.24 NTFS The NTFS file system is the recommended file system for use with Windows 2000. NTFS has all of the basic capabilities of FAT, and it provides the following advantages over the FAT and FAT 32 file systems: ⬤ Better file security. ⬤ Better disk compression. ⬤ Support for large hard disks, up to 2 terabytes (TB). (The maximum drive size for NTFS is much greater than that for FAT, and as drive size increases, performance with NTFS doesn't degrade as it does with FAT.) If you're using a dual-boot configuration (using both Windows 2000 and another operating system on the same computer), you may not be able to gain access to files on NTFS partitions from the other operating system on your computer. For this reason, you should probably use FAT32 or FAT if you want a dual-boot configuration. 13.25 Network Neighbourhood (My Network Places) In Windows 2000, Network Neighborhood has been renamed My Network Places and is located on the desktop. My Network Places provides a view of all the shared computers, files and folders, printers, and other resources on the network to which your computer is connected. A network is a group of computers connected to one another so that they can share resources such as files and printers. If your computer is connected to a network, you can use My Network Places to browse network resources the same way you browse the contents of your own computer. When you double-click My Network Places on your desktop, icons representing network resources appear in a window. Some of the following icons may appear in My Network Places. To open network resources 1. On the desktop, double-click My Network Places. 2. If you want to browse shared resources in your workgroup, double-click Computers Near Me. The computers and resources in your workgroup appear. If you want to view all of the resources available on the network, double-click Entire Network. Depending on how your network is set up, you may see computers arranged by domain or by workgroup. 3. Double-click the item you want to open, just as you would open drives and folders on your own computer 13.26 Sharing Information Between Programs – Concept of OLE- OBJECT LINKING AND EMBEDDING Cutting information from one document and pasting it into another is a good way of reusing information. When you paste information into a document, the information that you paste becomes part of the second document. If you make changes to the information in the original document, those changes will not be reflected in the fragment that you pasted into the second document. For Windows-based programs that support drag-and-drop functions, you can use linking or embedding to transfer information from one document to another document, even if that document is in a different program. Linking means inserting information that retains a connection to the information stored in the original file. Linking keeps the inserted information up-to-date by refreshing it when there is a change to the original document. To link selected information, use the Copy and Paste Special commands on the Edit menu. To link a file into your document, use the Object command on the Insert menu. For example, if you want to use WordPad to create text that you can reuse in other documents, you can use the Object command on the Insert menu to insert the text into your document as a linked WordPad file. Then, whenever you make changes to the text in the WordPad file, those changes automatically appear in any other documents that contain a link to that WordPad file. Some programs do not support linking. Embedding means inserting information (such as text or a graphic) into a document in a different program. The embedded information, or object, becomes part of the new document. Changes you make to the original document will not be reflected in the new document. To edit an embedded object, double-click it; the program in which the object was created opens. In order to edit an embedded object, the program in which the object was created (or another program capable of editing the object) must be installed on your computer. When you finish editing the object and return to your document, the document reflects all of the changes you've made. To embed selected information, use the Copy and Paste commands on the Edit menu. Use linking when you want the information to be updated dynamically as the data in the source file changes. Use embedding when you want to edit the information separately from the source file or if you think the source file will not be available. To embed information in another document 1. Select the information you want to embed in another document. 2. On the Edit menu, click Copy. 3. Click the location in the new document where you want to insert the information. 4. On the Edit menu, click Paste Special, click Paste, then click OK Notes ➢ Not all programs support embedding. If embedding isn't supported, your information will be copied. ➢ You can embed information in documents that were created in different programs. ➢ If you want the embedded information to be updated when it is changed in the original document, you may be able to link the information instead of embedding it. To copy information into another document 1. Select the information you want to copy. 2. On the Edit menu, click Copy. 3. In the document where you want the information to appear, click the location where you want to place the information. 4. On the Edit menu, click Paste. Notes ➢ You can also link or embed information. ➢ You can paste the information multiple times. To link information between documents 1. Select the information that you want to link to another document. 2. On the Edit menu, click Copy.

3. Click the location in the new document where you want to place the linked object. 4. On the Edit menu, click Paste Special. 5. Click Paste Link, then click OK. Notes ➢ Some programs do not support linking. If linking isn't available, Paste Specialwill be unavailable on the Edit menu. ➢ You can link information between documents that were created in different programs. To embed information in another document 1. Select the information you want to embed in another document. 2. On the Edit menu, click Copy. 3. Click the location in the new document where you want to insert the information. 4. On the Edit menu, click Paste Special, click Paste, then click OK Notes ➢ Not all programs support embedding. If embedding isn't supported, your information will be copied. ➢ You can embed information in documents that were created in different programs. ➢ If you want the embedded information to be updated whenit is changed in the original document, you may be able to link the information instead of embedding it. To link information between documents 1. Select the information that you want to link to another document. 2. On the Edit menu, click Copy. 3. Click the location in the new document where you want to place the linked object. 4. On the Edit menu, click Paste Special. 5. Click Paste Link, then click OK. Notes ➢ Some programs do not support linking. If linking isn't available, Paste Special will be unavailable on the Edit menu. ➢ You can link information between documents that were created in different programs Ways to share information in Application Programs- There are many different ways to share information between Office programs. You can make your

decision based on how you want the information to appear in the program, whether you want the information updated when it changes, and whom you want to share the information with. Summary ➢ MS-Windows 2000 is the most popular GUI for personal computers. Windows provides an environment that enhances DOS in many ways. ➢ Windows 2000 Professional offers increased compatibility with different types of networks and with a wide array of legacy hardware and software. ➢ Common Look and Feel, Device Independence, Device Independence, Memory Management, Support for existing DOS applications, Data Sharing, True 32-bit Operating System, Portability, Scalability, Multitasking, Multiple User Support, Multi-Threading, Graphical User Interface are important feature of windows OS. ➢ The desktop is the first view of the Windows 2000 operating system. The desktop contains icons, shortcuts and the taskbar. The user can navigate the user interface using the taskbar or the start menu. ➢ The start menu is a menu located at the left end of the taskbar. Start menu is integral to the clean look of the Windows NT desktop because it helps you to minimize the clutter on your desktop. ➢ Windows stores deleted files in the Recycle Bin. Which is located on the desktop. ➢ Control Panel is a convenient way to customize your computer. From Control Panel you can add and remove programs, fonts, and hardware; control how hardware, such as your mouse and modem, behaves; and select folder, display, and sound options. ➢ Windows Explorer displays the hierarchical structure of files, folders, and drives on your computer. It also shows any network drives which have been mapped to drive letters on your computer. ➢ WordPad is a text editor for short documents. Though, this accessory does not have the advanced features of full-blown word processors like MS-Word. You can format documents in WordPad with various fonts and paragraph styles. ➢ The WINDOWS 2000 components, accessories, or other applications can be added or installed, and removed by using the Add/Remove Programs utility from the Control Panel window. ➢ WINDOWS 2000 includes utilities to format diskettes, Check diskettes and hard drives for errors, defragments disks, and do other types of maintenance. ➢ When you cut or copy information from a program, it is moved to the Clipboard and remains there until you clear the Clipboard or until you cut or copy another piece of information.

Questions 1. Describe in brief the following desktop items. a. My computer b. Network neighborhood c. Recycle Bin d. Internet Explorer e. My Documents 2. How you will manage file and folder with windows explorer. 3. What is windows clipboard. Describe its usage and cut copy and paste operation. 4. Describe different components of windows operating. 5. How will you set up the DOS program in windows. 6. What are the multimedia devices work that work with windows. 7. What do you mean by term Plug and Play. 8. What is internet explorer, how do you enter the web addresses. 9. What is a taskbar in windows, how you will customize it. 10. What do you mean by OLE. 11. What is the use of control panel. 12. How will you install printer. 13. List various features of windows OS.

UNIT 14: Linux Operating System 14.0 Objectives 14.1 Introduction 14.2 Features of Linux Operating System 14.3 Advantages of Linux 14.4 Linux Structure 14.5 Various Flavours of Linux 14.6 Logging in and Logging out from a Linux Session 14.7 Starting a Linux Session: Logging in 14.8 Ending a Linux Session: Logging out 14.9 Types of Users in Linux 14.10 Security for Linux Users: Password 14.11 File System of Linux 14.12 File Naming Conventions in Linux 14.13 Relative Path Name 14.14 A Sample Directory Structure 14.15 Types of Files in Linux 14.16 Linux Commands 14.17 Vi Editor 14.18 The Insert Command and Replacement Command 14.0 Objectives After going through this unit to will be able to : ➢ Describe Hstory of linux, features and advantages of linux,versions of linux. ➢ Describe Structure of linux, login and logout, types of user and groups etc. ➢ Describe File system in linux, Directory structure etc ➢ Describe various commands of linux. 14.1 Introduction Linux Torvalds, a student at the University of Helsinki, Finland, introduced Linux in 1991.torvalds worked on the Linux project and wrote the source code for the Linux kernel. He made Linux available on the internet. Many programmers added to the code, changed it, and built in support for all kinds of hardware. In fact,

there are several versions of Linux available for different hardware platforms. Linux version 0.02 was introduced in 1991. Torvalds and hundreds of developers form across the world worked on it, and in March 1994, version 1.0 of the Linux kernel was released. Red hat Linux 6.0 uses version 2.2.5 of Linux kernel. Linux is a UNIX clone and has been written from scratch by Linux Torvalds. Torvalds was working on minix; a miniature version of UNIX, mainly used as a teaching aid in universities and colleges. He was impressed with the features of the Unix operating system. He wanted to create his own version of the UNIX operating system and give it freely for use to everybody. Since Linux is very similar to UNIX, it is important that you understand the origin and evolution of the UNIX operating system. UNIX started as a single user operating system. In 1969 ken, Thompson, with ideas and help for Dennis Ritchie and others, wrote a small general-purpose operating system for small machines. The operating system developed, attracted large number enthusiastic users, and was eventually developed to be a multi-user operating system. In 1973 Thompson and Ritchie rewrote the Unix Operating System in C, breaking away from the tradition of writing Operating Systemin assembly languages Around in 1974, UNIX was licensed to universities for educational purpose and few years later were made commercially available. It interesting to note that MS DOS was created much later than UNIX, by which time the industry has begun to accept UNIX as the standard Operating System. UNIX feature are therefore influenced the design of MS-DOS. Various version of UNIX are available in the industry, costing from a few thousand dollars to ten thousand dollars.Free clones are also available, such as the BSD (Berkeley software Distribution) Many vendors such as Sun, IBM, Hewlett-Packard, and others purchased the source code of the UNIX. They developed their own versions of UNIX. This resulted in a lot of difference in various versions of UNIX to avoid these confusions some standards ware maintained. Known as Portable Operating System for the UNIX environment (POSIX).POSIX was a set standard that enabled software has to run on different UNIX based Operating System without changing the source code. Linux is POSIX- complaint. Linux follows the open development model. Torvalds has made the source code of the LINUX kernel available for study and changes over the internet. This means that the current development version of LINUX is always open to everybody. Torvalds also accepts modifications to the kernel code. The result is that whenever a new Version o fix bugs, if any. In order to maintain stability, Torvalds exercises strict quality control and then merges all the new code into the kernel. This is in contrast to a closed model, in which a project term works on the software version and there are long intervals before people use the software and fix the bugs (errors) in it. This has been a major factor for the success of Linux. Linux has an official mascot- the Linux penguin, called, Tux, which was selected by Linux Torvalds to represent the image he associates with the operating system he created.

To help people gauge whether or not they are using a stable version of Linux, the following scheme is being used. In version 1. X Y, when x is an even number, it signifies a stable version of LINUX, whereas when x is an odd number it signifies a beta (trial) release of Linux. The internet has played a major role in the. Development of the Linux operation system. Today, many companies provide support for Linux over the internet. There are many Linux forums on the internet, and registration to these forums is free. You can subscribe to and get the latest information from these forums. 14.2 Features Of Linux Operating System Multi-programming Linux allows many programs to be executed simultaneously by different users. This feature is called multi-programming Time-sharing Multi-programming is made possible on the Linux system by the concept of time-sharing. The Operating System has to manage the various programs to be executed. The programs are queued and CPU time is shared among them. Each program gets CPU time for a specific period and is then put back in the queue to wait its turn again as the next program in the queue is attended to. Multi-Tasking A program in Linux is broken down into tasks, each task being something like reading from or writing to the disk or waiting for input from a user. The ability of any Operating System to handle the execution of multiple tasks is known as Multi-tasking. When a task is waiting for the completion of an activity, the CPU, instead of waiting time, starts executing the next task. Therefore, while one task is waiting for input from the user, another program could be reading from the hard disk. To explain the concept of multi-tasking, lets take a simple example. You are having a cup of coffee, reading a book, and talking to your friend over the phone. You are actually performing more than one task simultaneously. However, at a given point in time, you would be either sipping coffee, reading the book, or speaking over the phone. As you notice, you divide your time into smaller units and in each unit of time, you would be doing only one of the tasks. Similarly, the CPU divides the time between all the active tasks. The Kernel is responsible for scheduling the task. Virtual Memory The amount of physical memory may not always be sufficient for executing large application or having multiple application active at a point in time. In such situations, a partition can be utilized as the artificial

or the virtual memory. The system place the program and the data that are not frequently required in this portion of the hard disk and loads them in the memory whenever required. Thus, it use the resources of the computer to the fullest. Shared Libraries These are sets of functions or sub-routines maintained as a set of files. All the application that use these functions access them from the shared library file instead of individually maintaining code for them, thus saving hard disk space and memory. POSIX Compliance Linux is POSIX-compliance and support most of the standards set for all Unix system. Tomba The name Tomba is derived from the Server Message Block protocol or SMB. SMB is the protocol used by Microsoft Operating System to share file and print service Tomba is a suite of programs that implement the SMB protocol on Linux. Using Tomba, you can share a Linux file system with Windows 95, Windows 98, or Windows NT. You can also share a Windows 95,Windows 98,or Windows NT file system with Linux. SMB also enable the sharing of printer connected to either Linux or Windows 95, Windows 98,or Windows NT. Network Information service It is possible to share password and group files across a network in Linux using the Network Information service (NIS). NIS is a client and server database system. It is a central Database of account information that is used for account authentication. DOS Emulator Linux has a program called DOSEMU it creates a DOS emulator that can execute many DOS applications. DOSEMU use an image file of a hard disk to emulate a DOS hard drive so that you need not have a DOS partition.Linux can also use the DOS partition, if you have one on your system. CRON scheduler Linux has a scheduler program called CRON. It is used to run commands, scripts, or programs at schedule times.

Office Suites Linux supports many office suites like Applixware and Star Office. They can be used for reading and writing MS-Word and MS-Excel files. CurrentlyApplixware cannot be used for reading MS-Access databases. Applix Spreadsheet and Applix presentation graphics are similar to MS-Excel and MS-Power point. The latest version of Star Office is version 5. Data Archiving Utilities Linux provides utilities for basic data backup (archiving), such as tar, cpio, and dd. Red Hat Linux version 5.0 onward also provides a backup and Restore system unit (BRU), which can be purchased. BRU offer automated backup and scheduling. Licensing Linux is copyrighted under the GNU General Public License. The Licensing for Red Hat states that person can make any no of copies of the software and distribute it freely or charge a price for it. One can freely download Linux form the Internet for use. Web Server A Web server is the software that serves web pages. Linux comes with the Apache web server, which is the most popular web server is used today. Apache also supports the Squid proxy server that helps in improving the performance for accessing the Internet. Other Features Linux comes with a lot of other useful and free software such as text editors, browsers, and scientific applications. 14.3 ADVANTAGES OF LINUX Reliability : Linux is a stable Operating System. Linux server are not shut down for years together. This means that users on the Linux Operating System work consistently with the Linux server, without reporting any Operating System failure. Backward Compatibility:

Linux is said to be backward compatible. This implies that Linux has excellent support for older hardware. It can run on different types of processors, not on 386 and 486 Intel processor. It also runs well on DEC's Alpha processor, Sun's SPARK machine, Power and SGI MIPS GUI interface: A program interface that takes advantage of the computer's graphics capabilities to make the program easier to use. Well-designed graphical user interfaces can free the user from learning complex command languages. On the other hand, many users find that they work more effectively with a command-driven interface, especially if they already know the command language. In Linux GUI is divided into two sub-system consisting a server and a client. Linux has a number of GUI called Desktop Environment such as (KDE) K Desktop Environment and (GNOME) GNO Object Model Environment Multiple distributors There are many distributors of Linux. All the distribution use the Linux kernel, which is controlled by Linux Torvalds. These distributions add their own utilities and application and sell these as a customized package. Red Hat, Caldera, Mandrake, Debian, and Slack Ware. No known virus Linux is said to be free any virus attack. So far there are no virus for Linux Can support a high user load Linux can support a large no of user working simultaneously Development Libraries Linux offers an excellent platform for many development languages like C++ and Perl 14.4 Linux Structure Linux Operating System consist of three main software components as shown below 1. SHELL(KERNEL, HARDWARE) 2. UTILITIES 3. APPLICATION PROGRAM Components of the Linux Operating System KERNEL The core of the Linux system is the kernel - the Operating System program. The kernel controls the resources of the computer, allotting them to different user and task. It interacts directly with the hardware; the parts of the kernel must be customized to the hardware features of each system. However, the

kernel does not deal directly with a user. Instead of login process The central module of an operating system. It is the part of the operating system that loads first, and it remains in main memory. Because it stays in memory, it is important for the kernel to be as small as possible while still providing all the essential services required by other parts of the operating system and applications. Typically, the kernel is responsible for memory management, process and task management, and disk management. SHELL Linux has a simple user interface called shell that has power to provide the service that a user wants. It protects the user from having to know the intricate hardware details. 1) The outermost layer of a program. Shell is another term for user interface. Operating systems and applications sometimes provide an alternative shell to make interaction with the program easier. For example, if the application is usually command driven, the shell might be a menu-driven system that translates the user's selections into the appropriate commands. 2) Sometimes called command shell, a shell is the command processor interface. The command processor is the program that executes operating system commands. The shell, therefore, is the part of the command processor that accepts commands. After verifying that the commands are valid, the shell sends them to another part of the command processor to be executed. Linux Utilities and Application Programs The Linux utilities or commands are a collection of programs that service day to day processing requirements. The programs are invoked through the shell, which is itself another utility. Apart from the utilities provided as a part of Linux Operating System, more than a thousand Linux base application programs like database management system, word processor and various programs are available from independent software vendors 14.5 Various Flavors of Linux There are many distributors of Linux. All the distribution use the Linux kernel, which is controlled by Linux Torvalds. These distributions add their own utilities and application and sell these as a customized package. Red Hat, Debian, and Slack Ware are freely sable and once you buy them, you can sell them yourself. As a result, many other distributors take them as the base for adding utilities ultimately resulting in various flavour of Linux. Some of the popular distributors have been mentioned below. More information on the distributor can be found form their web site Red Hat

Red Hat is the most popular flavour of Linux. It is easy to install Red Hat Linux on your machine. It supports a wide variety of hardware as it has many driver for various hardware devices. the Red Hat 6.0 distribution also consist of an application CD with a fully functional Java Integrated Development Environment (IDE), a fully functional C++ IDE, and a standard C++ compiler. Red Hat also include Netscape Communicator as a part its distribution. Red Hat Linux is distributed by Red Hat, Inc. It is Y2K-compliant. Caldera The installation procedure of Caldera provides a wizard-like interface. It comes along with application software such as C++ compiler, Java IDE, and Netscape Communicator. It is Y2K-compliant. Caldera has two versions, open Linux 2.3, which ships latest Linux Kerneland has application like Star Office (private edition), and open Linux 1.3, which ships with Sybase ASE 11.0.3. Mandrake Mandrake is repackaging of Red Hat Linux; therefore, the flavour is always set behind the current Red Hat flavour. Debian Debian is among the oldest flavour of Linux. Since a term of volunteers in its developments, Debian has evolved slowly. This flavour utilize the Linux kernel but is designed to a accommodate other kernel as well. It is harder to setup than most other Linux version. Debian is commercial now. SuSE SuSE is a German flavour and it is quite similar to Caldera. There are a lot of application and utilities packed along SuSE Linux. It also contains Star Office (private edition) and word perfect (private edition). It is very popular in Europe. A lot of X development is being done on SuSE Slackware Slackware is one of popular flavour of Linux. Slackware version 3.6 has been found to be Y2K compliant but Slackware does not offer any Y2K guarantees 14.6 Logging in and Logging Out From a Linux Session Linux is an multi-user Operating System, it hold more than one computer with in the network and provides with various users to work in the Linux Operating System. A Linux machine is very secured it checks for every user for a password.

The process of connecting to the Linux sever with in the network is called Logging in and the process of disconnecting to the Linux server is called Logging out. A password it is allotted to the each and every user and every time a user logs in to the Linux server machine it has to pass to a security checking by checking the password of the user. If the password is not entered or if the password is incorrect the user is not allowed to login to the Linux machine. 14.7 Starting a Linux Session: Logging In A user of Linux-based system works at a user terminal. After you connect to the Linux system, a message similar to the one shown below appears at the terminal: Red Hat Linux release 6.0 (Hedwig) Kernel 2.2.5-15 on an i586 Login :_ Each user has an identification name called the user name or the login name, which has to be entered when the login : prompt appears. At the login : prompt after you enter your login name you are asked to enter the password. Linux keeps an record of the user name and information about the users in a special files. When you enter the login name and password, these are checked. If the login name and the password dose not match the user is not allowed to login to the Linux server When a valid user name is entered at the terminal, the [username@local host current directory name]$ symbol is displayed on the screen. The this is the shell prompt, in which username is the users login name and current directory _name is the user's login name and current directory name is the user's current working directory. The administration assign each user a HOME directory when a new login account is created. When you log in you are taken directly into your HOME directory. In Linux, login names (user names) are usually the name of user, and their HOME directory usually, although not necessary, has the tome name. For instance if the user name is 'tom' and your HOME directory name is also 'tom" after logging in, you will see the following prompt on the screen [tom@local host tom]$ You can now start working on Linux When you log on to Linux server, you have to be type the login name and password as these are case- sensitive. The entire login process appears the one shown below : Red Hat Linux release 6.0 (Hedwig) Kernel 2.2.5-15 on an i586 Login: tom Password : [user enters the password] Last login : Mon Oct 18 12:18:02 from 172.17.55.167 [tom@local host tom]$_ A sample of Linux Login Screen 14.8 Ending a Linux Session: Logging Out Once you have logged on to the system, your work session continues until you instruct the shell to terminate the session.

[tom@local host tom]$_ [tom@local host tom]$_ [tom@local host tom]$_ [tom@local host tom]$ logout Typing 'exit'or 'logout' at the command prompt ends your current Linux session The system then display login : prompt on the screen Red Hat Linux release 6.0 (Hedwig) Kernel 2.2.5-15 on an i586 Login:_ In order to maintain the security of file, you should never leave the terminal without logging out. 14.9 Types of Users in Linux System Administrator The system administrator (SA) is primarily responsible for the smooth operation of the system. It is the SA's job to switch on the system console (the machine on which the operating system resides, also known as the server machine). The SA also creates users and groups of users for the system and takes backups to prevent loss of data due to system breakdown. In LINUX the SA is also known as the root user. The SA has all the rights for the LINUX system. File Owner The user who creates a file is said to be the owner of that file. The owner of a file can perform any operation on that file copying, deleting, and editing. Group Owner Consider the following situation: A project team of five people from the Dynasoft Consultants Inc. is working on a software development project for private detective agency. An analyst heads the team. The other four members are programmers. The team is working on a LINUX system. Each programmer has been given a few programs to develop. The data provided by the detective agency is of a highly confidential nature, and so the data file has been created in the analyst's HOME directory. One programmer may have to ;link (join) a program to another programmer's program in order to test5 the program. In this situation, each programmer is the file owner of his or her own program files. Each program, however, also belongs to the other programmers, so that they can use it for linking to the file or directly access the file. The project team of five users is said to be the group owner for the file. In LINUX, it is possible to ;define the users who will belong to a group. A group of users area also given a name, just as a user is given a name. Other Users In the example of the Dynasoft Consultants Inc., all the users of the system who are not members of the project group are referred to as Other Users for the files of that group. other users are users who do not belong to the particular group. For example, the users belonging to the finance department

could be treated as other users for the payroll department. 14.10 Security For Linux User: Passwords The procedure followed by users, to start a LINUX session ensures that only authorized users can access the system. However, it is easy for anyone to find out your login name since it is displayed on screen when being entered. Thus, the login procedure is not a foolproof system. To overcome this problem LINUX allows an additional measure of security by allowing you to have a password associated with your login name. to; log in using a password protected user name, you have to enter not only the user name but also the password. The combination of the two is checked by Linux to see if it is an authorized entry. The Linux prompt appears only if you have entered both the user name and the password correctly, otherwise you are asked to re-enter the user name followed by the password. Passwords are not displayed on the screen while they are being entered. Changing the User Password A user can also change his password with the password command. The steps followed by the user. Steve, to change his password are depicted below: Example. [steve@localhost Steve] $ passwd changing password for Steve (current) UNIX password: [user enters old password hear] new UNIX password: [user enters new password here] retype new UNIX password: [user re-enters new password here] password: all authentication tokens updated successfully [Steve@ local host Steve] $ - The password command asks for the old password to ensure that only the authorized user is trying to change the password. In case LINUX does not recognize the old password, it displays a 'passwd: authentication failure ' message, and the LINUX prompt appears 14.11 File System Of Linux A single fixed disk can store thousand of file. For organizing data by grouping file on disk, the Operating System provides a file system. When Unix was developed, it has features that the other Operating System at that time did not have. One such feature was its file system. The Unix file system has a hierarchical structure and files can be stored under directories. Directories are similar to the drawers of a filing cabinet just as drawers contains files each of which contain document of a similar nature, so also the directories on the disk can be created to store files containing data of a similar nature. Just as in a filing cabinet, the user decides the drawer labels and contents, so also the user decide the directory name and the files in the directory. Linux follows the Unix filesystem convention. FIG :- A SAMPLE LINUX FILE HIERARCHY All the files are stored on the disk under one main directory called the root directory. The root directory has been further sub divided into directories - bin, boot, home, usr, etc, and dev. In each directory, files containing related data can be stored. The administrator of the Linux system may place all the HOME directories of the user under the /home directory. For example, the tom directory is the HOME directory for the user, tom. he will store all his file in his home directory, tom, or create new directory under it.. The dir tom contain two file namely program 1 and program 2, and two dir DESKTOP AND TEMPLATE. the desktop dir contain two file namely, default and start.

In the filesystem hierarchy, a file is referred to by its path name, which is made up of the file name, preceded by the name of the directory containing the file. The path name may have a set of directories, one directory containing another, until the '/' directory is reached. The file name and the different directory names contained in the path are separated by the '/' symbol. For instance, in the previous figure, the full path of the file 'start' can be given as : /home/tom/ desktop/start. Similarly, the path for the file 'ls' is /bin /ls. As you notice, the '/' is a special character, therefore, it cannot be included in a file or directory name. In a hierarchical, inverted tree-like structure, the Operating System provides faster access to files because group of files are isolated from each other. Only one directory has to be searched to locate a file. On the other hand, if there were no sub-directories and all the file were stored under the root directory, the locating a file would have meant searching through the entire list of files till the required files were located. This feature of Linux - a hierarchical inverted tree file system - is also found in MS-DOS. There is however, a major difference between the MS-DOS and Linux file system. Linux comprises hundreds of files that are provided along with the Linux- based system, while MS-DOS consist of only three files. When you install Linux, it creates a number of directories and sub-directories to store the Operating System files. Thus, unlike MS-DOS, Linux enforces the hierarchical filesystem. MS-DOS is much smaller than Linux in term of storage requirements. A hard disk created for use on MS-DOS has only one directory: the root directory. A hard disk created for use on Linux has a much larger hierarchical structure. The following are some of the directories under the / directory. Each of these directories has been organized to store a specific type of file The / bin directory stores many utilities of Linux. These utilities are the commands available under the Linux system. These are in the binary format, hence the name "bin". The /dev directory stores all the device-related file for the system. The /etc directory stores the system-related data which user and the system need to refer to, such as the passwd file. These are generally the essential miscellaneous system program and configuration files. The /lib directory contains libraries of data for the compilers installed in the system for Example, the C language routine. The /home directory generally contains all the HOME directory of users. The /user directory store the Operating System file, which are not involved in the boot process. Many utilities are available in the bin directory under the user directory. The /user /bin directory is different from the /bin directory. The /var directory has information specific to different utilities of Linux.

14.12 File Naming Conventions In Linux It can be up to 256 characters long It can contain special characters, except for '/' It can contain both upper-case and lower-case alphabets Are case sensitive It should not have a blank or a tab 14.13 Relative Path Name The current working directory is the directory in which you are currently located. You can refer to is as you current directory or your current working directory. While referring to a file, instead of specifying the full path of the file, you can specify the path in relation to your current directory. Consider the following sample directory structure. A Sample Directory Structure For instance, let's say that the user, Steve, is working in the directory, ./home/Steve. Wants to access the file, jan99, under the /home/stave/baseball directory. If he accesses this file using the full path name, Steve would specify the path as /home/stave/baseball /jan99. Alternatively, Steve can specify the relative path, baseball/jan99. This is a much more convenient way of specifying the path name. Thus, notice that when you start the path name with a ' /?' ' LINUX Takes the path name as an absolute path name or the full path name. Whenever the path name starts with character, it is taken as the relative path with respect to the current working directory. Referring to your HOME Directory Each user on the linux system is assigned a directory to store files. After logging in, a user is directly taken to the user's HOME directory. In other words, the working directory of the user after logging in to the LINUX system is his or her HOME directory. While working on linux, you may often feel the need to access the HOME directory in a convenient way. In LINUX, you can specify the path for your HOME directory with the tilde ( ) sign. For instance, if Steve wants to access the Jan 99 file, he can specify? /baseball/jan99 from anywhere in the files?the file name with the path as system. 14.15 Types of Files in Linux IN LINUX all information is treated as a file. So, besides a user's program files and data files, there are also special files, such as those that contain information about directory contents to the various input/output devices connected to the system. In LINUX, a device is also treated as a file and all the information going to, say, the VDU (Visual Display Unit) is treated as if it were being sent to a file. In LINUX there, are three categories of files: (1) Ordinary files (2) Directory files (3) Special files Ordinary Files All the files created by a user come under this category of files. These include all the data files, program files, object files, and executable files. A user can make changes to such files. Directory Files LINUX automatically creates a directory file when a directory is created. This file has the same name as the directory, and contains information about the files under the directory. For example for the directory, /home/Steve, there will be a directory file called Steve in the directory../home, which contains information on all the files and directories under the directory. Steve. Adirectory file cannot be modified by a user but is, instead, modified automatically by the system when a new file or a subdirectory is added to the directory. Special Files Most of the system files in LINUX are special files. Special files are typically associated with input/output devices and are found in the standard LINUX directories, such as/dev and /etc. Users cannot alter special files. LINUX supports the following types of special files: (1) Character device files (2) Block device files (3) Hard links (4) Symbolic links

Character device files read and write data one character at a time. A modem is an example of a character device file. Character devices are also called sequentially accessed devices. Block device files can access a block of data at a time. generally, a block of data can be either 512 or 1.024 bytes. Here, the kernel reads or writes minimum one block of data at a time. the kernel collects the data in the memory for the amount of block memory and then makes it available to a user. Such devices allow random access. This makes input/output operations faster. An example of a block device is the hard disk. Many of the devices can act as character devices or block devices depending on the command that is used to access the device. Hard links are special files that allow a single file to have multiple names. You can have a hard link file only for a file and not for a directory. These links are known as hard links because they create a direct link to an inode. Since each file system has its own inode information database, you can hard-link files only when they're on the same file system symbolic links (see below) don't have this restriction. Symbolic links also known as soft links, are similar to hard links except that they can work across different file systems. FIGURE: FILE TYPES IN LINUX 14.16 LINUX COMMANDS BC The BC utility can be used as a calculator. It can perform calculations on huge decimal values. The bc utility takes entries from a file that has mathematics expression and display the result of each expression on the standard output. If you do not. CAL

The cat (concatenate) command display the contents of the specified file . The cat command can be used to verify the contents of more than one file. Example ------< [sam@localhost sam]$ cat data (Press Enter) a sample file [sam@localhost sam]$_

The command assumes that the file data is in the current directory. Complete path name can also be specified to display a file in another directory. The cat command can also display more than one file as shown in the following command: Example ----- < [sam@localhost sam]$ cat data bata (Press Enter) a sample file another sample command [sam@localhost sam]$ _ CHGRP This command is used to change the group for a user Example ----- < [sam@localhost sam]$ chgrp user /home/steve (Press Enter) will change the group to sam CHOWN This command is used to set the ownership of the directory or a file Example ----- < [sam@localhost sam]$ chown aaa /home/Steve (Press Enter) CHMOD The access permission associated with a file or directory can be changed using the chmod command. Only the owner of a file can change the permission

associate with it. Syntax ----- < chmod mode file/s You can specify the mode (permission for a type of user) in symbols by using the numbers (the absolute way). The symbolic format uses letters to represent permissions and the absolute way use the numeric format. Symbolic mode In the symbolic mode, the permission and the type of user for whom the permission is to be granted in symbols. The table below states the symbols for different permissions Symbol Meaning R Read W Write

X Execute The following tables states the symbolic representation of entities Symbol Meaning U Owner of the file or directory G Member of the same group O All the other users A All users To change the permission for a file, the owner has to specify : The type of user for whom the permission is to be changed The type of permission that is to be changed Whether the permission is to be given or invoked The name of the file for which the permission is to be changed Example ----- < [sam@localhost sam]$ ls-l abc.c(Press Enter) -rw-rw-r-- 1 Steve IRDTECH 50 Sep 22:21:01 abc.c [sam@localhost sam]$ chmod u+x abc.c (Press Enter) Here 'u' indicates the file owner, '+' indicates that the permission is to be given 'x' indicates execute permission and abc.c is the file name The file access permission for the file abc.c will now appear as [sam@localhost sam]$ ls-l abc.c(Press Enter) -rwxrw-r-- 1 Steve IRDTECH 50 Sep 22:21:01 abc.c Absolute Mode The absolute mode specifies FAPs uses a series of digits to represent the actual permissions. It is a more convenient method since it can specify different permission for all type of users in one statement. The following table number for each type of permission Number Permission 4 Read 2 Write 1 Execute

Absolute number for permission The figure below depicts a sample representation of absolute numbers. r w x r w - r - - Sum is 7 Sum is 6 Sum is 4 Given below is a table that shows the different permission that are allocated after a particular absolute value is used with chmod command. Mode used with the chmod command Permission of the file after the command Meaning 400 -r--------- Only the read permission to the owner 6 --------rw- Only the read and write permission to the user 61 ----rw---x The execute permission to the other user and the read and write permission to group members 747 -rwxr x The read write and execute permission to the owner, only the read permission to group users and only the execute permission to other users. 777 -rwxrwxrwx All the permissions to all the users Mode and Output chmod command CLEAR The clear command is used to clear the terminal screen. Syntax---------< clear (Press Enter) COPY The cp (copy) command duplicates the contents of the contents of the source file into a target file. Syntax ------< cp[option]>source file/s<>destination file/s< Example ------< [sam@localhost sam]$ cp data bata (Press Enter)

In the above example the contents of data are copied to a new file bata. If the bata already exist its contents will be overwritten by the contents of data Complete path name can be specified with the cp command to copy file across the directories. You can also copy a directory recursively using the cp command with the -r option Example ------< [sam@localhost sam]$ cp -r temp tempo (Press Enter) The above command copies the temp directory and all its sub-directory to tempo directory. If the tempo directory exist all the contents are put inside that directory, otherwise tempo is created in the current working directory. The other options and their functions are Options Function -I Prompts before overwriting -l Links a file instead of coping it -s Creates a symbolic link -v Verbose-explain what is being done, in detail DATE All the Linux system display the current date and time. User can display the current date and time using the date command Example-----< [sam@localhost sam]$ date (Press Enter) FRI APR 21:36:42 EST 2002 [sam@localhost sam]$ cp data bata (Press Enter) Option Function %m Month of the year in digit %d Day of the month in digit %y Years (last two digit) %D Date as mm/dd/yy %H Hour (00 to 23) %M Minutes (00 to 59)

%S Seconds (00 to 59) %T Time as HH:MM:SS %a Abbreviated weekdays (Sun to Sat) %h Abbreviated month (Jan to Dec) %r Time in the AM/PM notation Example ----- < [sam@localhost sam]$ date "+%T" (Press Enter) 22:47:45 [sam@localhost sam]$ date "+%y" (Press Enter) 01 LS (List) The ls command is used to display the name of directory and files. It is very much similar to DIR command in DOS. [sam@localhost sam]$ls /home/sam (Press Enter) AAA x ball comm bat Desktop a.out bball program.cc This command will display the files and directory in the sam directory. In the above example the output the list file is displayed but no file type is displayed. The -l option use with ls displays the detailed list of files and directory. Example ----- < [sam@localhost sam]$ls-l (Press Enter) total 12 -rw-rw-r-- 1 sam sam 134 Dec 21 20:18 desktop drw--w-r-- 3 sam sam 1034 Dec 23 10:18 a.out -r---w-r-- 2 sam sam 341 Dec 13 10:00 program.cc -rw--w-r-- 1 sam sam 234 Dec 23 00:18 a.out --w--w-r-- 2 sam sam 434 Dec 20 20:15 a.out -r--dw-r-- 1 sam sam 124 Dec 11 21:12 a.out dr---w-r-- 1 sam sam 103 Dec 12 22:10 a.out drw-rw-r-- 1 sam sam 114 Dec 21 20:11 a.out drw--w-rw- 2 sam sam 156 Dec 20 12:20 a.out drw--w-r-- 1 sam sam 111 Dec 22 21:00 a.out

drwxrwr-x 2 sam sam 158 Dec 21 10:20 a.out Explanation of the output of ls - l command is Column No Explanation 1 File type and File Access Permission (FAPs) 2 Number of links 3 File owner 4 Group owner 5 File size 6 Day 7&8 Time of last modification 9 Name of file The characters in column on denote the file type. The following table provide an explanation Symbol File Type Symbol File

Type - Ordinary file D Directory B Block special file C Character special file L Symbolic link P Named pipe S Symbolic link The other options available with ls command are as follows Option Function Option Function -a Lists all the files including the hidden files -F Show the file type along with the name ("/"for a directory, '*' for executable) -R Dose a recursive listing, that is, displays the contents of the specified directory and sub- directory -r Displays the directory and sub - directory in reverse order -S Sorts list by file size -A Displays the file of almost all directory except. and.. directory Common option of the ls Command Example ----- < [sam@localhost sam]$ls-al (Press Enter)

total 22 -rw-rw-r-- 1 sam sam 134 Dec 21 20:18 desktop -rw-rw-r-- 1 root root 224 Dec 21 10:18 top drw--w-r-- 3 sam sam 1034 Dec 23 10:18 a.out -r---w-r-- 2 sam sam 341 Dec 13 10:00 program.cc -rw--w-r-- 1 sam sam 234 Dec 23 00:18 a.out -rw--w-r-- 2 sam sam 434 Dec 20 20:15 tap -rw-rw-r-- 1 sam sam 134 Dec 21 20:18 desktop -rw-w-r-- 2 sam sam 341 Dec 13 10:00 program.cc -r--dw-r-- 1 sam sam 124 Dec 11 21:12 a.out dr---w-r-- 1 sam sam 103 Dec 12 22:10 one drw-rw-r-- 1 sam sam 114 Dec 21 20:11 a.out -rw-rw-r-- 1 sam sam 134 Dec 21 20:18 ppp drw--w-rw- 2 sam sam 156 Dec 20 12:20 a.out drw--w-r-- 1 sam sam 111 Dec 22 21:00 a.out z-rw-rw-r-- 1 sam sam 134 Dec 21 20:18 aaa drwxrwr-x 2 sam sam 158 Dec 21 10:20 a.out [sam@localhost sam]$_ MAIL The mail utility can be used to send and receive mail. It uses the ed editor to compose mail. The mail utility is used for sending message in a script. You can get information about the mail utility from the manual page. MKDIR(Make Directory) The MKDIR (make directory) command is used to creare a directory. Example ------< [Steve@localhost Steve]$ mkdir aaa (Press Enter) [Steve@localhost Steve]$ _ The sub-directory aaa, is crated under the current directory. MORE

The cat command is used to display the contents of a file

on the screen. If the file being displayed is large, then the entire contents will scroll up the screen. To view file on the screen - full at a time you can the more command. The more command is used to display data one screen-full at a time. While viewing the file using the

more command, once you have scrolled down you cannot move up. Syntax ----- < more [option]>file name< Example ----- < [sam@localhost sam]$ more aaa (Press Enter) The above command will display a page-wise listing of contents of file aaa MV (Move) The mv (move) command is used to a file directory form one location to another or to change the name of a file or directory. Note that moving a file from location to anther is different from copying a file. While moving a file no new file is created. Syntax ----- < mv [option] source destination Example ----- < [sam@localhost sam]$ mv com computers (Press Enter) The above command will rename the com to computers Example < [sam@localhost sam]$ mv data3 /home/Steve/program/data3 (Press Enter) In the above example the file data3 is moved from the current directory to the /home/Steve/ program directory. Some of the options available with the mv command are Options Functions -f Removes the existing destination by force -I Interactively, prompts before overwriting at the destination location -v Verbose- explains what is being done, in detail PWD (present working directory) ------< Display the full path name of the currently working directory. Example------< [Steve@localhost Steve]$ pwd (Press Enter) /home/Steve [Steve@localhost Steve]$ _ here /home/Steve is the current directory. RM The rm (remove) command is used to delete file or directories. Syntax ---- < rm [option] file/s (Press Enter) Example ---- < [Steve@localhost Steve]$ rm data1 data2 (Press Enter) the above command will remove the file data1 and data2 from your current directory. [Steve@localhost Steve]$ rm /home/Steve/data1 (Press Enter) the above command will delete the data1 from the specified path RMDIR (remove directory) ------< The RMDIR(remove directory) command removes the directory specified. Example < [Steve@localhost Steve]$ rmdir aaa (Press Enter) [Steve@localhost Steve]$ _ the specified directory 'aaa' will be deleted complete path name may also be specified with RMDIR [Steve@localhost Steve]$ rmdir /home/Steve/aaa (Press Enter) [Steve@localhost Steve]$ _ WC The wc is a filter command which counts the number of line, words and characters in a disk file or in the standard input. Syntax ----- < wc[-lwc][file name/s] Example-------------< [Steve@localhost Steve]$ wc aaa (Press Enter) 5 14 129 [Steve@localhost Steve]$ _ The file aaa has 5 lines, 7 words, and 29 characters. The following table summarizes the options of the wc filter. Options Functions -l Displays the number of lines -w Displays the number of words -c Displays the number of characters Options of the wc filter WHO The who command is used to display the name of all the users who are currently logged in. Example ------ < [Steve@localhost Steve]$ who (Press Enter) root tty1 Sep 21 12:29

root tty2 Sep 21 14:44 Steve tty3 Sep 21 11:19 sam pts/0 Sep 21 11:15 (172.12.55.178) tom pts/1 Sep 21 12:29 (172.12.55.133) root pts/2 Sep 21 16:05 (172.12.55.136) The following table gives the description of the various columns

Write The write utility is used to write text on another user's terminal. The syntax of the write utility is the same as that of the talk utility (this is not true if you are using the remote host version of talk) Syntax ------< write user_name [terminal_name] the write utility will allow text another user's terminal. Example ------< [Steve@localhost Steve]$ write tom (Press Enter) this command will display the following massage on tom's terminal message from Steve@linuxmac on pts/1 at 13:30.... 14.17 Vi EDITOR Editor is an text editor used to create and manage text file and documents. An editor is application software, which is usually bundled with an Operating System. A text editor has variety of applications such as Creating and maintaining documents Writing programs and utilities Writing mail messages The vi editor is a visual editor which is use for editing text file creating and editing text file containing data, documents, or programs. It display the contents of file on the screen and allows a user to add, insert, delete, or change parts of the text. It is amongst the oldest editor available on the Linux platform. The vi editor is invoked by using following command at the Linux prompt : Vi filename (Press Enter) Example ----- < [root@localhost Steve]$vi mess.txt (Press Enter)

if the file exists the editor will display the file on the screen using the editor else will create a new file by the specified name The vi editor works in two mode the insert mode and the command mode. In the insert mode, you can add contents to a file. To move from the insert mode to the command mode, you have to press the >Esc< key. The vi editor has different commands to add, change and delete text. Command Action i Inserts text at cursor position The command shown in the table above a user to start entering text. To end the insertion or the append mode, you have to press the >Esc< key. This takes you back to the command mode. The Command in Vi editor Refers to the table given below for some of the commonly used command in vi Command Action H Move to previous character L Mover up to next character K Move up one line J Move down one line X Delete character at current cursor location Dd Delete line :wq (Press Enter) Saves all changes and quit :w (Press Enter) Save the file :q! (Press Enter) Quit without changing :e >file name< (Press Enter) Open file specified :w >file name< (Press Enter) Writes to different file :w! >file name< (Press Enter) Force write to another file :! >command_name< (Press Enter) Executes a shell command THE CURSOR MOVEMENT COMMAND IN VI The basic cursor movement command are shown in the table below Command Action >ctr<d or D Scrolls down half screen >ctr<u or U Scrolls up half screen >ctr<F Page forward >ctr<B Page backward

0 zero Beginning of line $ End of line THE INSERT COMMAND AND REPLACE COMMAND Refers to the table below for the insert and replace command Command Action A Appends after current character A Appends at the end of the line I Insert before the current character I Insert at beginning of the line O Inserts a blank line below and allows insertion O Inserts blank line above and allow insertion Rx Replaces current character with character x Rtext>Esc< Replace character with text (until >Esc< is pressed. That is it takes you to the replace mode instead of append mode Summary ➢ Linux torvalds worked on the Linux project and wrote the source code for the Linux kernel. ➢ Multi-programming, Time-sharing, Multi-Tasking, Virtual Memory, Shared Libraries, POSIX Compliance, Tomba, Network Information service, DOS Emulator are features

of linux. ➢

The kernel controls the resources of the computer, allotting them to different user and task. It interacts directly with the hardware; the parts of the kernel must be customized to the hardware features of each system. ➢ Linux has a simple user interface called shell that has power to provide the service that a user wants. It protects the user from having to know the intricate hardware details. ➢ Variour flavor of linux are RedHat, Caldera, Mandrake, Debian, SuSE, Slackware, fedora etc. ➢ Linux is an multi-user Operating System, it hold more than one computer with in the network and provides with various users to work in the Linux Operating System. A Linux machine is very secured it checks for every user for a password. ➢ The process of connecting to the Linux sever with in the network is called Logging in and the process of disconnecting to the Linux server is called Logging out. ➢ File Owner The user who creates a file is said to be the owner of that file. The owner of a file can perform any operation on that file copying, deleting, and editing. ➢ In LINUX, it is possible to define the users who will belong to a group. A group of users area also

given a name, just as a user is given a name. ➢ In linux all the files are stored on the disk under one main directory called the root directory. The root directory has been further sub divided into directories - bin, boot, home, usr, etc, and dev. ➢ The vieditor is a visual editor which is use for editing text file creating and editing text file containing data, documents, or programs. It display the contents of file on the screen and allows a user to add, insert, delete, or change parts of the text. Questions 1. What is Linux? why it is popular? 2. Explain the login and logout process of Linux. 3. What do you mean by multi user operating system? 4. Explain the file system of Linux. 5. What is Kernel? 6. What is shell? 7. How many types of user and groups are there in Linux? 8. What is the use of chmod command in Linux? 9. Explain cat, copy, md and mv command in Linux. 10. Write a short note on Vi Editor.

Hit and source - focused comparison, Side by Side

| 1/158 | SUBMITTED TEXT | 11 WORDS | 100% | MATCHING TEXT | 11 WORDS |

and every calculation is performed with the same accuracy. The

and every calculation is performed with the same accuracy. The

W https://pdfcookie.com/documents/36746371-foundations-of-computing-pradeep-k-sinha-bpb-publication …

| 2/158 | SUBMITTED TEXT | 14 WORDS | 100% | MATCHING TEXT | 14 WORDS |

They are also known as peripheral devices because they surround the CPU and

They are also known as peripheral devices because they surround the CPU and

W https://pdfcookie.com/documents/36746371-foundations-of-computing-pradeep-k-sinha-bpb-publication …

| 3/158 | SUBMITTED TEXT | 16 WORDS | 70% | MATCHING TEXT | 16 WORDS |

calculations are to be performed, a computer will perform every calculation with the same accuracy.

calculations have to be performed, a computer will perform the ten millionth calculation with exactly the same accuracy

W https://pdfcookie.com/documents/36746371-foundations-of-computing-pradeep-k-sinha-bpb-publication …

| 4/158 | SUBMITTED TEXT | 25 WORDS | 66% | MATCHING TEXT | 25 WORDS |

It rolls on a small ball and has two or three buttons on the top. When you roll the mouse across a flat surface

It rolls on a small bearing and has one or more buttons on the top. When a user rolls the mouse across a flat surface,

W https://pdfcookie.com/documents/36746371-foundations-of-computing-pradeep-k-sinha-bpb-publication …

| 5/158 | SUBMITTED TEXT | 12 WORDS | 100% | MATCHING TEXT | 12 WORDS |

These answer sheets are directly fed to a computer for grading

These answer sheets are directly fed to a computer for grading

W https://pdfcookie.com/documents/36746371-foundations-of-computing-pradeep-k-sinha-bpb-publication …

| 6/158 | SUBMITTED TEXT | 16 WORDS | 84% | MATCHING TEXT | 16 WORDS |

approach of printing, printers are classified as impact and non-impact printers. Impact printers use the

Approach of printing. According to this classification, printers are classified as impact printers and non-impact printers. Impact printers use the

W https://pdfcookie.com/documents/36746371-foundations-of-computing-pradeep-k-sinha-bpb-publication …

| 19/158 | SUBMITTED TEXT | 26 WORDS | 78% | MATCHING TEXT | 26 WORDS |

is made up of several small storage locations (ICs) called cells. Each of these cells can store a fixed number of bits called word length.

is made up of several small storage areas called locations or cells. Each of these locations can store a fixed number of bits called word length

W https://pdfcookie.com/documents/36746371-foundations-of-computing-pradeep-k-sinha-bpb-publication ...

| 20/158 | SUBMITTED TEXT | 15 WORDS | 100% | MATCHING TEXT | 15 WORDS |

memory is like a large cabinet containing as many drawers as there are addresses

memory is like a large cabinet containing as many drawers as there are addresses

W https://pdfcookie.com/documents/36746371-foundations-of-computing-pradeep-k-sinha-bpb-publication ...

| 21/158 | SUBMITTED TEXT | 18 WORDS | 87% | MATCHING TEXT | 18 WORDS |

memory. Each drawer contains a word and the address is written on outside of the drawer.

memory. Each drawer contains a word and the address of each word is written on the outside of the drawer.

W https://pdfcookie.com/documents/36746371-foundations-of-computing-pradeep-k-sinha-bpb-publication ...

| 22/158 | SUBMITTED TEXT | 12 WORDS | 100% | MATCHING TEXT | 12 WORDS |

the computer uses a number of special memory units called registers.

the computer uses a number of special memory units called registers.

W https://pdfcookie.com/documents/36746371-foundations-of-computing-pradeep-k-sinha-bpb-publication ...

| 23/158 | SUBMITTED TEXT | 14 WORDS | 95% | MATCHING TEXT | 14 WORDS |

temporarily and pass it on as directed by the control unit. 2.1

temporarily, and to pass it on as directed by the control unit.

W https://pdfcookie.com/documents/36746371-foundations-of-computing-pradeep-k-sinha-bpb-publication ...

| 24/158 | SUBMITTED TEXT | 24 WORDS | 93% | MATCHING TEXT | 24 WORDS |

Secondary storage devices need a controller to act as an intermediary between the device and the rest of the computer system. In some

Secondary storage devices need a controller to act as an intermediary between the device and the rest of the computer system. On some

W https://pdfcookie.com/documents/36746371-foundations-of-computing-pradeep-k-sinha-bpb-publication ...

Keyboard Keyboard devices are the most commonly used input devices today. They allow data entry into a computer system by pressing a set of keys (labeled button), which are neatly mounted on a keyboard,

Keyboard Devices Keyboard devices are the most commonly used input devices today. They allow data entry into a computer system by pressing a set of keys (labeled buttons), which are neatly mounted on a keyboard

W https://pdfcookie.com/documents/36746371-foundations-of-computing-pradeep-k-sinha-bpb-publication …

Mouse As of today, mouse is the most popular point-and-draw device. It has become a must-have input device on personal computer and workstations, which have a GUI-based user interface. Amouse is a small hand-held device, which can comfortably fit in a user's palm. It rolls on a small bearing, and has one or more buttons on the top. When a user rolls the mouse across a flat surface, such as on top the table on which the computer is placed, the graphics cursor moves on the screen of the video display terminal in the direction of the mouse's movement. The graphics cursor, which is used to point and draw, is displayed on the screen as a variety of symbols, such as an arrow, a wrist with pointing finger, etc. Depending on the application, the text and graphics cursor may be displayed on the screen at the same time. Science all movements of the mouse are reproduced by the graphics cursor on the screen, you can move the graphics cursor at a menu item or icon by moving the mouse. For example, move the mouse away from you to move the cursor up on the screen, move the towards you to move the cursor down on the screen, move the mouse right to move the cursor right on the screen, and move the mouse left to move the cursor left on the screen. In this way, the mouse can be used for quick positioning of the graphics cursor on the desired menu item or icon. When positioned at a menu item or an icon, the graphics cursor is said to point to that menu item or icon. With a click of the mouse's button, the system can then be notified of this choice. Note that, notifying the system of a particular choice, out of the various options provided by the software, is much easier with a mouse than pressing various key combinations. With the proper software, a mouse can also be used to draw picture on the screen and edit text. 4.4

Mouse As of today, the mouse is the most popular point-and-draw device. It has become a musthave input device on personal computers and workstations that have a GUI-based video display terminal. As shown in Figure 9.4, the mouse is a small hand-held device, which can comfortably fit in a user's palm. It rolls on a small bearing and has one or more buttons on the top. When a user rolls the mouse across a flat surface, such as on top of the table on which the computer is placed, the graphics cursor moves on the screen of the video display terminal in the direction of the mouse's movement. The graphics cursor, which is used to point and draw, is displayed on the screen as a variety of symbols such as an arrow (fl), a wrist with a pointing finger (m~), etc. Depending on the application, the text and graphics cursor may be displayed on the screen at the same time. Since all movements of the mouse are reproduced by the graphics cursor on the screen, you can move the graphics cursor at a menu item or an icon by moving the mouse. For example, move the mouse away from you to move the cursor up on the screen, move the mouse towards you to move the cursor down on the screen, move the mouse right to move the cursor right on the screen, and move the mouse left to move the cursor left on the screen. In this way, the mouse can be used for quick positioning of the graphics cursor on the desired menu item or an icon. When positioned at a menu item or an icon, the graphics cursor is said to point to that menu item or icon. With a click of the mouse's button, the system can then be notified of this choice. Note that notifying the system of a particular choice out of the various options provided by the software is much easier with a mouse than pressing various key combinations. With the proper software, a mouse can also be used to draw pictures on the screen and edit text.

W https://pdfcookie.com/documents/36746371-foundations-of-computing-pradeep-k-sinha-bpb-publication …

SUBMITTED TEXT 111 WORDS **91% MATCHING TEXT** 111 WORDS

Trackball A trackball is a pointing device, which is similar to a mouse. As shown in the ball, which is placed in the base of a mouse, is placed on the top along with the buttons, in case of a trackball. To move the graphics cursor around the screen, the ball is rolled with the fingers. Because the whole device is not be moved for moving the graphics cursor, a trackball requires less space than a mouse for operation. Since it need not be moved for moving the graphics cursor, it is often attached to or built into the keyboard. Track ball built into the keyboard are commonly used in

Trackball A trackball is a pointing device that is similar to a mechanical mouse. As shown in Figure 9.5, the ball that is placed in the base of a mechanical mouse is placed on the top along with the buttons in case of a trackball. To move the graphics cursor around the screen, the ball is rolled with the fingers. Because the whole device is not moved for moving the graphics cursor, a trackball requires less space than a mouse for operation. Since it need not be moved for moving the graphics cursor, it is often attached to or built into the keyboard. Trackballs built into the keyboard are commonly used in

w https://pdfcookie.com/documents/36746371-foundations-of-computing-pradeep-k-sinha-bpb-publication …

SUBMITTED TEXT 40 WORDS **100% MATCHING TEXT** 40 WORDS

laptop (notebook) computers, because a mouse is not practical for laptop users in a small space. Because the ball is fixed on the top, some people find it helpful to think of a trackball as an

laptop (notebook) computers because a mouse is not practical for laptop users in a small space. Because the ball is fixed on the top, some people find it helpful to think of a trackball as an

w https://pdfcookie.com/documents/36746371-foundations-of-computing-pradeep-k-sinha-bpb-publication …

SUBMITTED TEXT 91 WORDS **97% MATCHING TEXT** 91 WORDS

A trackball comes in various shapes and forms with the same functionality. The three commonly used shapes are a ball, and a square. In case of a ball, the ball is rolled with the help of fingers to move the graphics cursor. In case of a button, the button is pushed with a finger in the desired direction of the graphics cursor movement. In case of a square plastic, the fingers is placed on top of it, and moved in the desired direction of the graphics cursor movement. 4.5

A trackball comes in various shapes and forms with the same functionality. The three commonly used shapes are a ball, a button, and a square. In case of a ball, the ball is rolled with the help of fingers to move the graphics cursor. In case of a button, the button is pushed with a finger in the desired direction of the graphics cursor movement. In case of a square plastic, the finger is placed on top of it and moved in the desired direction of the graphics cursor movement.

w https://pdfcookie.com/documents/36746371-foundations-of-computing-pradeep-k-sinha-bpb-publication …

SUBMITTED TEXT 49 WORDS **97% MATCHING TEXT** 49 WORDS

Touch Screen Touch screen is the most simple, intuitive, and easiest to learn of all input devices. A touch screen enables the users to choose from available options by simply touching with their finger the desired icon or menu item displayed on the computer's screen. Touch screen

Touch Screen Touch screen is the most simple, intuitive, and easiest to learn of all input devices. A touch screen enables the users to choose from available options by simply touching with their finger the desired icon or menu item displayed on the computer's screen. Computers with touch screen

w https://pdfcookie.com/documents/36746371-foundations-of-computing-pradeep-k-sinha-bpb-publication …

Bar Code Reader: Data coded in the form of small lines (known as bars) are known as bar codes. Bar codes represent alphanumeric data by a combination of adjacent vertical lines (bars) by varying their width and the spacing between them. They are particularly used for unique identification of all types of goods, books, postal packages, badges, tags, etc. A bar-code reader is a device, which is used for reading (decoding) bar-coded data. It may be a hand- held scanner, or may be embedded in a stationary scanner. It scans a bar-code image, and converts it into an alphanumeric value, which is then fed to the computer to which the bar-code reader is connected, just as though the alphanumeric value had been typed on a keyboard. A bar-code reader uses a laser-beam scanning technology. The laser beam is stroked across the pattern of bars of a bar code. Different bar codes, having different patterns of bars, reflect the beam in different ways, which is sensed by a light-sensitive detector. Reflected light patterns are converted into electrical pulses, and then transmitted to recognition logic circuits, which convert it into an alphanumeric value. Just as there are a variety of internal bit encoding systems, there are a variety of bar-coding systems. The most widely known bar-coding system is the Universal Product Code (UPC), which now appears on almost all retail packages in USA. The UPC, originally used for supermarket items, is now being printed on other consumer goods also. The UPC bar-code patterns are decoded as 10 digits. The first 5 of these digits identify the manufacturer or supplier of the product, and the next 5 digits identify a specific product of the manufacturer. Bar code readers are commonly found in supermarkets and department stores. When a customer picks up a product for purchasing, and brings it to the cash counter for payment, the sales person at the counter uses a bar-code reader to scan the bar code printed on the product. The bar-code

Bar-Code Reader Data coded in the form of small lines (known as bars) are known as bar codes. Bar codes represent alphanumeric data by a combination of adjacent vertical lines (bars) by varying their width and the spacing between them. They are particularly used for unique identification of all types of goods, books, postal packages, badges, tags, etc. A bar-code reader is a device used for reading (decoding) bar-coded data. It may be a hand-held scanner, or may be embedded in a stationary scanner. It scans a bar-code image and converts it into an alphanumeric value, which is then fed to the computer to which the bar-code reader is connected, just as though the alphanumeric value had been typed on a keyboard. A bar-code reader uses a laser-beam scanning technology. The laser beam is stroked across the pattern of bars of a bar code. Different bar codes having different patterns of bars reflect the beam in different ways, which is sensed by a light-sensitive detector. Reflected light patterns are converted into electrical pulses and then transmitted to recognition logic circuits which convert it into an alphanumeric value. Just as there are a variety of internal bit encoding systems, there are a variety of barcoding systems. The most widely known bar-coding system is the Universal Product Code (UPC), which now appears on almost all retail packages in USA. The UPC, originally used for supermarket items, is now being printed on other consumer goods also. The UPC bar-code patterns are decoded as 10 digits. The first 5 of these digits identify the manufacturer or supplier of the product and the next 5 digits identify a specific product of the manufacturer (see Figure 9.13). Bar code readers are commonly found in supermarkets and department stores. When a customer picks up a product for purchasing and/brings it to the cash counter for payment, the sales person at the counter uses a barcode reader to scan the bar code printed on the product. The bar-code reader converts the bar code

W   https://pdfcookie.com/documents/36746371-foundations-of-computing-pradeep-k-sinha-bpb-publication ...

SUBMITTED TEXT   91 WORDS   **98%**   **MATCHING TEXT**   91 WORDS

reader converts the bar code into an alphanumeric value, and feeds it to the computer, which looks up the price of the product, possibly updates inventory and sales records, and uses the price and description information to prepare a purchase bill for the customer. Postal delivery services also use bar codes to identify and track each packet. For example Federal Express of USA uses a unique bar-coding system, and their employees can usually tell a customer, within a matter of minutes, the current location of any packet. 4.7

reader converts the bar code into an alphanumeric value and feeds it to the computer that looks up the price of the product, possibly updates inventory and sales records, and uses the price and description information to prepare a purchase bill for the customer. Postal delivery services also use bar codes to identify and track each packet. For example, Federal Express of USA uses a unique bar-coding system and their employees can usually tell a customer within a matter of minutes the current location of any packet.

W   https://pdfcookie.com/documents/36746371-foundations-of-computing-pradeep-k-sinha-bpb-publication ...

SUBMITTED TEXT   31 WORDS   **85%**   **MATCHING TEXT**   31 WORDS

Data scanning devices are input devices, which are used for direct data entry into the computer system from source documents. Some of the data scanning devices are also capable of

Data scanning devices are input devices used for direct data entry into the computer system from source documents. Some of the data scanning devices are also capable of

W   https://pdfcookie.com/documents/36746371-foundations-of-computing-pradeep-k-sinha-bpb-publication ...

SUBMITTED TEXT   33 WORDS   **95%**   **MATCHING TEXT**   33 WORDS

recognizing marks or characters. Data scanning devices typically have the following characteristics: ➢ They eliminate some of human effort required to get data into the computer. Human beings do not have to

recognizing marks or characters. Data scanning devices typically have the following characteristics: 1. They eliminate some of the duplication of human effort required to get data into the computer. Human beings do not have to

W   https://pdfcookie.com/documents/36746371-foundations-of-computing-pradeep-k-sinha-bpb-publication ...

SUBMITTED TEXT   72 WORDS   **97%**   **MATCHING TEXT**   72 WORDS

the data. ➢ The reduction in human intervention improves data accuracy and can increase the timeliness of the information processed. ➢ Science scanners are direct data entry devices, they demand high quality of input documents. Documents that are poorly typed or have strikeovers or erasures are normally rejected. ➢ With these devices, form design and ink specification may become more critical than is the case when people key

the data. 2. The reduction in human intervention improves data accuracy and can increase the timeliness of the information processed. 3. Since scanners are direct data entry devices, they demand high quality of input documents. Documents that are poorly typed or have strikeovers or erasures are normally rejected. 4. With these devices, form design and ink specification may become more critical than is the case when people key

W   https://pdfcookie.com/documents/36746371-foundations-of-computing-pradeep-k-sinha-bpb-publication ...

scanner. As shown in a flatbed scanner is like a copy machine, which consists of a box having a glass plate on its top and a lid that covers the glass plate. The document to be scanned is placed upside down on the glass plate and moves horizontally from left to right when activated. After scanning one line the light beam moves up a little and scans the next line. The process is repeated for all the lines. It takes about 20 seconds to scan documents of size 21cm *28cm. 2. Hand-held scanner. A hand-held scanner has a set of light emitting diodes encased in a small case, which can be conveniently held in hand during operation. To scan a document, the scanner is dragged from one end of the document to its other end, with its light on. The scanner has to be dragged very steadily and carefully over the document; otherwise, the conversion of the document into its equivalent bit map will not be correct. Due to this reason, hand-held scanners are used only in cases where high accuracy is not needed

scanner. As shown in Figure 9.8, flat-bed scanner is like a copy machine consisting of a box having a glass plate off its top and a lid that covers the glass plate. The document to be scanned is placed upside down on the glass plate. The light source is situated below the glass plate and moves horizontally from left to right when activated. After scanning one line, the light beam moves up a little and scans the next line. The process is repeated for all the lines. It takes about 20 seconds to scan a document of size 21 cm x 28 cm. 2. Hand-held scanner. As shown in Figure 9.9, a hand-held scanner has a set of light emitting diodes encased in a small case that can be conveniently held in hand during operation. To scan a document, the scanner is slowly dragged from one end of the document to its other end with its light on. The scanner has to be dragged very steadily and carefully over the document, otherwise the conversion of the document into its equivalent bit map will not be correct. Due to this reason, hand-held scanners are used only in cases where high accuracy is not needed.

is low. They are also much cheaper as compared to flatbed scanners. 4.8 Optical

is low. They are also much cheaper as compared to flat-bed Optical

Optical Mark Reader (OMR) These scanners are capable of recognizing a pre-specified type of mark made by pencil or pen. For example, many students might have appeared in objective type tests,

Optical Mark Reader (OMR) These scanners are capable of recognizing a pre-specified type of mark made by pencil or pen. For example, many students might have appeared in objective type tests

they had to mark their answers to questions on a special pre-printed test scoring sheet by darkening a small square, circular, or oval shaped box by a pencil, to indicate their correct choice out of the various given options. These answer sheets are directly fed a computer for grading with the use of an optical mark reader. The actual technique used by an OMR device for recognition of marks involves focusing a light on the page being scanned and detecting the reflected light pattern from the marks. Pencil marks made with a soft lead pencil (high graphite content) reflect the light, enabling the reader to determine which responses are marked. 4.9

they had to mark their answers to questions on a special pre-printed test scoring sheet by darkening a small square, circular or oval shaped box by a pencil to indicate their correct choice out of the various given options (see Figure 9.12). These answer sheets are directly fed to a computer for grading with the use of an optical mark reader. The actual technique used by an OMR device for recognition of marks involves focusing a light on the page being scanned and detecting the reflected light pattern from the marks. Pencil marks made with a soft lead pencil (high graphite content) reflect the light enabling the reader to determine which responses are marked.

handwritten), they have the following two limitations: ➢ Since the input document is stored as an image, instead of text, it is not possible to do any word processing of the document (the computer cannot interpret the stored document as letters, numbers and special characters). ➢ The storage required for storing the document as an image is much more than that required

handwritten), they have the following two limitations: 1. Since the input document is stored as an image instead of text, it is not possible to do any word processing of the document (the computer cannot interpret the stored document as letters, number and special characters). 2. The storage required for storing the document as an image is several times more than that required

storing the same document as a text. For example, a page of printed text, having 2000 characters, can be stored as 2000 bytes

storing the same document as a text. For example, a page of printed text having 2000 characters can be stored as 2000 bytes

using the ASCII representation. A bit map image representation of the same document will require 10 to 15 times more storage, depending on the resolution (granularity) of grid points. The OCR technology is used to overcome these limitations. In this case, the scanner is equipped with a character recognition software (called OCR software),

using the ASCII representation. A bit map image representation of the same document will require 10 to 15 times more storage depending on the resolution (granularity) of gri points. The OCR technology is used to overcome these limitations. In this case, the scanner is equipped with a character recognition software (called OCR software)

SUBMITTED TEXT 39 WORDS **92%** **MATCHING TEXT** 39 WORDS

converts the bit map images of characters to equivalent ASCII codes. That is, the scanner first creates the bit map image of the document, and then the OCR software translates the array of grid points into ASCII text,

converts the bit map images of characters to equivalent ASC codes. That is, the scanner first creates the bit map image of the document and then the OCR software translate the array of grid points into ASCII text

W https://pdfcookie.com/documents/36746371-foundations-of-computing-pradeep-k-sinha-bpb-publication ...

SUBMITTED TEXT 14 WORDS **100%** **MATCHING TEXT** 14 WORDS

the computer can interpret as letters, numbers and special characters.

the computer can interpret as letters, numbers and special characters.

W https://pdfcookie.com/documents/36746371-foundations-of-computing-pradeep-k-sinha-bpb-publication ...

SUBMITTED TEXT 18 WORDS **100%** **MATCHING TEXT** 18 WORDS

OCR software is extremely complex, because it is difficult to make a computer recognize an unlimited number

OCR software is extremely complex, because it is difficult to make a computer recognize an unlimited number (

W https://pdfcookie.com/documents/36746371-foundations-of-computing-pradeep-k-sinha-bpb-publication ...

SUBMITTED TEXT 32 WORDS **91%** **MATCHING TEXT** 32 WORDS

typefaces and fonts. Hence, these software are designed to recognize texts, which are written using standard type fonts (called OCR fonts). Two such standard fonts are OCR-A(American standard) and OCR-B(European standard).

typefaces and fonts. Hence these software are designed to recognize texts that are written using standard type. fonts (called OCR fonts). Two such standard fonts are OCR-A (American standard) and OCR-B (Europea standard).

W https://pdfcookie.com/documents/36746371-foundations-of-computing-pradeep-k-sinha-bpb-publication ...

SUBMITTED TEXT 30 WORDS **94%** **MATCHING TEXT** 30 WORDS

document contains italics or bold face letters, or fonts other than that for which the OCR software has been designed, the OCR software will not work effectively. 4.10

document contains italics or bold face letters, or fonts other than that for which the OCR software has bee designed, the OCR software will not work effectively.

W https://pdfcookie.com/documents/36746371-foundations-of-computing-pradeep-k-sinha-bpb-publication ...

SUBMITTED TEXT 21 WORDS **85%** **MATCHING TEXT** 21 WORDS

recognition devices are input devices, which allow a person to input data to a computer system by speaking to it.

recognition devices are input devices that allow a person to input data to a computer system b) speaking to it.

W https://pdfcookie.com/documents/36746371-foundations-of-computing-pradeep-k-sinha-bpb-publication ...

it. Hence, they make computers much easier to use. However, as a data input device, currently voice recognition systems have limited success, because correct interpretation by a machine of the large number of words in the vocabulary of a language is difficult. The major difficulty has been that people speak with different accents (pronounce differently) and intonations (with different tone or pitch of the voice), and the fact that the meaning of words can vary depending on the context in which they are used. Hence, today's voice recognition systems are limited to accepting few words within a relatively small domain and can be used to enter only limited kinds and quantities of data.

it. Thus they make computers much easier to use. However as a data input device, currently voice recognition systems have limited success because correct interpretation by a machine of the large number of words in the vocabulary of a language is difficult. The major difficulty has been that people speak with different accent (pronounce differently) and intonations (with different tone or pitch of the voice), and the fact that the meaning of words can vary depending on the context in which they are used. Hence today's voice recognition systems are limited to accepting few words within a relatively small domain and can be used to enter only limited kinds and quantities of data.

w https://pdfcookie.com/documents/36746371-foundations-of-computing-pradeep-k-sinha-bpb-publication ...

Although in its infancy, voice recognition systems are already being used for a wide range of applications. Some of its typical applications are as follows: ▪ For inputting data to a computer system by a person in situations where his/her hands are busy, or his/her eyes must be fixed on a measuring instrument or some other object. For example,

Although in its infancy, voice recognition systems are already being used for a wide range of applications. Some of its typical applications are as follows: 1. For inputting data to a computer system by a person in situations where his/her hands are busy, or his/her eyes must be fixed on a measuring instrument or some other object For example,

w https://pdfcookie.com/documents/36746371-foundations-of-computing-pradeep-k-sinha-bpb-publication ...

in an operation room can request certain information about a patient while operating. ▪ For authentication of a user by a computer system based on voice input. ▪ For limited use of computers by individuals with physical disabilities. In addition to making input of data easier, voice recognition systems also provide tremendous freedom of movement to the operator, because the operator is free to stand up and move around, while inputting voice data to the system. 4.11 Video-Input System (

in an operation room can request certain information about a patient while operating. 2. For authentication of a user by a computer system based on voice input. 3. For limited use of computers by individuals with physical disabilities. In addition to making input of data easier, voice recognition systems also provide tremendous freedom of movement to the operator because the operator is free to stand up and move around while inputting voice data to the system. Vision-System

w https://pdfcookie.com/documents/36746371-foundations-of-computing-pradeep-k-sinha-bpb-publication ...

SUBMITTED TEXT 171 WORDS **98% MATCHING TEXT** 171 WORDS

System (Vision-Input System) A vision-input system allows a computer to accept input just by seeing an object. The input data in this case is normally an object's shape and features in the form of an image. The idea is to simulate the capability of a human vision system in a limited sense. A computer with a vision-input device consists of a digital camera, and follows the following steps to recognize a given object: ➢ The digital camera is focused on the input object to take a picture of the object. ➢ The digital camera creates an image of the object in digital form (in 1s and 0s), s.o that it can be stored and interpreted by the computer. ➢ The digitized image of the object is matched against similarly formed pre-stored images in the computer's image database. ➢ Depending on whether a match is found or not, the system takes the appropriate action. Vision-input devices are mainly used today in factories for designing industrial robots,

system. Vision-Input System A vision-input system allows a computer to accept input just by seeing an object. The input data in this case is normally an object's shape and features in the form of an image. The idea is to simulate the capability of a human vision system in a limited sense. A computer with a vision-input device consists of a digital camera and follows the following steps to recognize a given object: 1. The digital camera is focused on the input object to take a picture of the object. 2. The digital camera creates an image of the object in digital form (in 1s and 0s) so that it can be stored and interpreted by the computer. 3. The digitized image of the object is matched against similarly formed pre-stored images in the computer's image database. 4. Depending on whether a match is found or not, the system takes the appropriate action. Vision-input devices are mainly used today in factories for designing industrial robots

SUBMITTED TEXT 46 WORDS **93% MATCHING TEXT** 46 WORDS

are used for quality" control and assembly processes. For example, a robot used for quality control may inspect objects and separate those, which do not meet certain quality-control specifications. Another robot may be used to separate objects of different shapes in different bins. 4.12

are used for quality-control and assembly processes. For example, a robot used for quality-control may inspect objects and separate those, which do not meet certain qualitycontrol specifications. Another robot may be used to separate objects of different shapes in different bins.

SUBMITTED TEXT 15 WORDS **100% MATCHING TEXT** 15 WORDS

Printers Printers are the most commonly used output devices today for producing hard-copy output.

Printers Printers are the most commonly used Output devices today for producing hard-copy output.

SUBMITTED TEXT 20 WORDS **100% MATCHING TEXT** 20 WORDS

The various types of printers in use today are described below. Dot-Matrix Printers Dot-matrix printers are character printers,

The various types of printers in use today are described below. Dot-Matrix Printers Dot-matrix printers are character printers

are normally slow with speeds usually ranging between 30 to 600 characters per second. However, they are cheap in terms of both initial cost and cost of operation. Hence, they are preferred by individuals and organizations for generating printed outputs, if the speed and quality of printing are not important factors. They are also used for applications, which require multicopy output, such as shipping forms and invoices, which rely on impact for generating multiple copies.

are normally slow with speeds usually ranging between 30 to 600 characters per second. However they are cheap in terms of both initial cost and cost of operation. Hence they are preferred by individuals and organizations for generating printed outputs if the speed and quality of printing are not important factors. They are also used for applications that require multicopy output such as shipping forms and invoices, which rely on impact for generating multiple copies.

Inkjet Printers Inkjet printers are character printers, which form characters and all kinds of images by spraying small drops of ink on to the paper. The print head of an inkjet printer contains up to 64 tiny nozzles, which can be selectively heated up in a few microseconds by an integrated circuit register. When the register heats up, the ink near it vaporizes, and is ejected through the nozzle, and makes a dot on the paper placed in front of the print head. To print a character, the printer selectively heats the appropriate set of nozzles as the print head moves horizontally. Inkjet printers produce higher quality output than dot-matrix printers, because they form characters by very tiny ink dots, A high-resolution inkjet printer has as many as 64 nozzles within height of 7mm, providing print resolution of around 360 dots per inch. Since inkjet printers produce printed output as patterns of tiny dots, they can print any shape of characters, which a programmer can describe. This allows the printer to print many special characters, different sizes of print, and the ability to print graphics, such as charts and graphs.

InkJet Printers InkJet printers are character printers that form characters and all kinds of images by spraying small drops of ink on to the paper. The print head of an inkjet printer contains up to 64 tiny nozzles, which can be selectively heated up in a few micro seconds by an integrated circuit register. When the register heats up, the ink near it vapourizes and is ejected through the nozzle and makes a dot on the paper placed in front of the print head. To print a character, the printer selectively heats the appropriate set of nozzles as the print head moves horizontally. An inkjet printer is shown in InkJet printers produce higher quality output than dot-matrix printers because they form characters by very tiny ink dots. A high-resolution inkjet printer has as many as 64 nozzles within height of 7mm, providing print resolution of around 360 dots per inch. Since inkjet printers produce printed output as patterns of tiny dots, they can print any shape of characters that a programmer can describe. This allows the printer to print many special characters, different sizes of print, and the ability to print graphics such as charts and graphs.

Inkjet printers are non-impact printers, because they print by spraying ink on the paper. Hence, they are quiet in operation. However, like other non-impact printers, inkjet printers cannot be used to produce multiple copies of a document in a single printing. Inkjet printers

InkJet printers are non-impact printers because they print by spraying ink on the paper. Hence they are quiet in operation. However, like other non-impact printers, inkjet printers cannot be used to produce multiple copies of a document in a single printing. InkJet printers

are slower than dot-matrix printers with speeds usually ranging between 40 to 300 characters per second. Typically, an inkjet printer is more expensive than a

are slower than dot-matrix printers with speeds usually ranging between 40 to 300 characters per second. Typically, an inkjet printer is more expensive than a

W https://pdfcookie.com/documents/36746371-foundations-of-computing-pradeep-k-sinha-bpb-publication …

They are preferred by individuals and organizations for generating printed outputs, if the speed of printing is not an important factor.

They are preferred by individuals and organizations for generating printed outputs if the speed of printing is not an important factor.

W https://pdfcookie.com/documents/36746371-foundations-of-computing-pradeep-k-sinha-bpb-publication …

Laser Printers Laser printers are page printers, which print one page at a time. The main components of a laser printer are a laser beam source, a multi-sided mirror, a photoconductive drum and toner (tiny particles of oppositely charged ink). To print a page of output, the laser beam is focused on the electro statically charged drum by the spinning multi-sided mirror. The mirror focuses the laser beam on the Surface of the drum in a manner to create the patterns of characters/images to be printed on the page. As the drum is photoconductive, a difference in electric charge is created on those parts of the drum surface, which are exposed to the laser beam. Asa result, the toner, which is composed of oppositely charged ink particles, sticks to the drum in the places, where the laser beam has charged-the drum's surface. The toner is then permanently fused on the paper with heat and pressure to generate the printed output. The drum is then rotated and cleaned with a rubber blade to remove the toner sticking to its surface to prepare the drum for the next page printing. Fig. 11 : Laser

Laser Printers Laser printers are page printers that print one page at a time. The main components of a laser printer are a laser beam source, a multi-sided mirror, a photoconductive drum and toner (tiny particles of oppositely charged ink). To print a page of output the laser beam is focused on the electro statically charged drum by the spinning multi-sided mirror. The mirror focuses the laser beam on the surface of the drum in such a manner that it creates the patterns of characters/images to be printed on the page. As the drum is photoconductive, a difference in electric charge is created on those parts of the drum surface that are exposed to the laser beam. a result, the toner, which is composed of oppositely charged ink particles, sticks to the drum in the places the laser beam has charged the drum's surface. The toner is then permanently fused on the paper with heat and pressure to generate the printed output. The drum is then rotated and cleaned with a rubber blade to remove the toner sticking to its surface to prepare the drum for the next page printing. A laser

W https://pdfcookie.com/documents/36746371-foundations-of-computing-pradeep-k-sinha-bpb-publication …

Laser printers produce very high quality output, because they form characters by very tiny ink particles. The most common laser printers have resolution of 600 dpi (

Laser printers produce very high quality output because they form characters by very tiny ink particles. The most common laser printers have resolutions of 600 dpi,

W https://pdfcookie.com/documents/36746371-foundations-of-computing-pradeep-k-sinha-bpb-publication …

produce horizontal motion. The movements of the drum and the pen(s) are controlled by the graph-plotting program. That is, under computer control, the drum and the pen(s) move simultaneously to draw the designs and graphs on the sheet placed on the drum. The plotter can also annotate the designs and graphs so drawn by using the pen to draw characters of various sizes. Since each pen is program selectable, pens having ink of different colors can be mounted in different holders to produce multi- colored designs. Flatbed Plotter A flatbed plotter plots a design or graph on a sheet of paper,

produce horizontal motion. The movements of the drum and the pen(s) are controlled by the graph plotting program. That is, under computer control, the drurn and the pen(s) move simultaneously to draw the designs and graphs on the sheet placed on the drum. The plotter can also annotate the designs and graphs so drawn by using the pen to draw characters of various sizes. Since each pen is program selectable, pens having ink of different colors can be mounted in different holders to produce multi-colored designs. A drum plotter is shown in Figure 9.34. Flatbed Plotter I A flatbed plotter plots a design or graph on a sheet of paper

that is spread and fixed over a rectangular flatbed table. In this type of plotter, normally the paper dose not move, and the pen holding mechanism is designed to provide all types of motions necessary to draw complex designs and graphs. That is, under computer control, the pen(s) move in the required manner to draw complex designs and graphs on the sheet placed on the flatbed table. The plotter can also annotate the designs and graphs so drawn by using the pen to draw characters of various sizes. Here also, prevision is there to mount more than one pen in the pen(s) holding mechanism. Since each pen is program selectable, pens having ink of different colors can be mounted in different holders to produce multi-colored designs. The plot size is restricted by the area of the bed. Some may be as small as A4 size (8"*11"page), while some very large beds used in the design of cars, ships, aircrafts, buildings, highways, etc. can be up to 20 ft. by 50 ft. some plotters are also designed to etch plastic or metal plates. In this case, the plastic or metal sheet is spread on the bed, and the drawing pen has a sharp-edged needle.

that is spread and fixed over a rectangular flatbed table. In this type of plotter, normally the paper does not move and the pen holding mechanism is designed to provide all types of motions necessary to draw complex designs and graphs. That is, under computer control, the pen(s) move in the required manner to draw the designs and graphs on the sheet placed on the flatbed table. The plotter can also annotate the designs and graphs so drawn by using the pen to draw characters of various sizes. Here also is there to mount more than one pen in the pen(s) holding mechanism. Since each pen is program selectable, pens having ink of different colors can be mounted in different holders to produce multi colored designs. The plot size is restricted by the area of the bed. Some may be as small as. A4 size (8"xll" page) while some very large beds used in the design of cars, ships, aircrafts, buildings, highways, etc. can be up to 20 ft. by 50 ft. Some plotters are also designed to etch plastic or metal plates. In this case, the plastic or metal sheet is spread on the bed and the drawing pen has a sharp-edged needle.

Non-Positional Number System In this system there are symbols used as I for 1, II for 2, III for 3

non-positional number system. In this system, we have symbols such as I for 1, II for 2, III for 3,

98/158	SUBMITTED TEXT	20 WORDS	86%	MATCHING TEXT	20 WORDS

Computer circuits only have to handle two binary digits, this greatly simplifies the internal circuit design of computers. ➤

computer circuits only have to handle two binary digits rather than ten decimal This greatly simplifies the internal circuit design of computers,

W https://pdfcookie.com/documents/36746371-foundations-of-computing-pradeep-k-sinha-bpb-publication ...

99/158	SUBMITTED TEXT	19 WORDS	100%	MATCHING TEXT	19 WORDS

Everything that can be done in decimal number system can also be done in binary number system.

everything that can be done in decimal number system can also be done in binary number system.

W https://pdfcookie.com/documents/36746371-foundations-of-computing-pradeep-k-sinha-bpb-publication ...

100/158	SUBMITTED TEXT	16 WORDS	86%	MATCHING TEXT	16 WORDS

Number System Usually numbers expressed in decimal number system are much more meaningful to us,

NUMBER SYSTEM TO Numbers expressed in decimal number system are much more meaningful to us

W https://pdfcookie.com/documents/36746371-foundations-of-computing-pradeep-k-sinha-bpb-publication ...

101/158	SUBMITTED TEXT	26 WORDS	78%	MATCHING TEXT	26 WORDS

we have been using decimal numbers in our daily routine life. Any number in one number system can be represented in any other number system.

we have been using decimal numbers in our day-today life right from childhood. any number value in one number system can be represented in any other number system.

W https://pdfcookie.com/documents/36746371-foundations-of-computing-pradeep-k-sinha-bpb-publication ...

102/158	SUBMITTED TEXT	17 WORDS	84%	MATCHING TEXT	17 WORDS

There are many methods, which can be used to convert numbers from one base to another.

There are many methods techniques that can be used to convert numbers from one base to another.

W https://pdfcookie.com/documents/36746371-foundations-of-computing-pradeep-k-sinha-bpb-publication ...

103/158	SUBMITTED TEXT	54 WORDS	87%	MATCHING TEXT	54 WORDS

Determine the column(positional) value of each digit. This depends on the position of the digit and the base of the number system. ➤ Multiply the obtained column values by the digits in the corresponding columns. ➤ Sum the products calculated in the above step. The total is the equivalent value in decimal.

Determine the column (positional) value of each digit (this depends on the position of the digit and the base of the number system). Multiply the obtained column values (in Step 1) by the digits in the corresponding columns. Sum the products calculated in Step 2. The total is the equivalent value in decimal.

W https://pdfcookie.com/documents/36746371-foundations-of-computing-pradeep-k-sinha-bpb-publication ...

109/158	SUBMITTED TEXT	30 WORDS	75% MATCHING TEXT	30 WORDS

that the maximum value of one digit is equal to the maximum value of three digits in binary. Therefore, the value of one octal digit is equivalent to 3

that the maximum value of one digit in hexadecimal is equal to the maximum value of four digits in binary. Thus, the value range of one digit of hexadecimal is equivalent to

w https://pdfcookie.com/documents/36746371-foundations-of-computing-pradeep-k-sinha-bpb-publication ...

110/158	SUBMITTED TEXT	30 WORDS	80% MATCHING TEXT	30 WORDS

that the maximum value of one digit is equal to the maximum value of four digits in binary. Therefore, the value of one hexadecimal digit is equivalent to 4

that the maximum value of one digit in hexadecimal is equal to the maximum value of four digits in binary. Thus, the value range of one digit of hexadecimal is equivalent to

w https://pdfcookie.com/documents/36746371-foundations-of-computing-pradeep-k-sinha-bpb-publication ...

111/158	SUBMITTED TEXT	17 WORDS	81% MATCHING TEXT	17 WORDS

algebra, which deals with the binary number system. It is very useful in designing logic circuits,

algebra that deals with the binary number system, is very useful in designing logic circuits

w https://pdfcookie.com/documents/36746371-foundations-of-computing-pradeep-k-sinha-bpb-publication ...

112/158	SUBMITTED TEXT	25 WORDS	81% MATCHING TEXT	25 WORDS

the processors of computer systems. Boolean algebra provides an economical and straightforward approach to the design of relay, and other types of switching circuits.

the design of modern computers. Boolean algebra provides an economical and straightforward approach to the design of relay and other types of switching circuits.

w https://pdfcookie.com/documents/36746371-foundations-of-computing-pradeep-k-sinha-bpb-publication ...

113/158	SUBMITTED TEXT	24 WORDS	94% MATCHING TEXT	24 WORDS

Boolean algebra is used extensively in designing the circuitry used in computers. 6.9.1 Fundamental concepts of Boolean Algebra ➢

Boolean algebra is now being used extensively in designing the circuitry used in computers. Fundamental Concepts of Boolean Algebra

w https://pdfcookie.com/documents/36746371-foundations-of-computing-pradeep-k-sinha-bpb-publication ...

The procedure of Boolean operators is as follows:- ➢ The expression is scanned from left to right. ➢ Expressions enclosed within parenthesis are evaluated first. ➢ All compliment (NOT) operations are performed next. ➢ All "." (AND) operations are performed after that. ➢ Finally all "+" (OR) operations are performed in the end. 6.9.2

The precedence of Boolean operators is as follows: 1. 2. 3. 4. 5. The expression is scanned from left to right. Expressions enclosed within parentheses are evaluated first. All complement (NOT) operations are performed next. All '•' (AND) operations are performed after that. Finally, all'+' (OR) operations are performed in the end.

Postulates of Boolean Algebra Postulate 1: (a) A=0 if and only if A is not equal to 1 (b) A=1 if and only if A is not equal to 0 Postulate 2: (a) x + 0 = x (b) x.1 = x Postulate 3: Commutative Law (a) x + y = y + x (b) x.y = y.x Postulate 4: Associative Law (a) x + (y + z) = (x + y) + z (b) x.(y.z) = (x.y).z Postulate 5: Distributive Law (a) x.(y + z) = x.y + x.z (b) x + y.z = (x + y).(x + z) Postulate 6: (a) x + x' = 1 (b) x.x' = 0 These postulates are the basic axioms of the algebraic structure, which need no proof. They are used to prove the theorems of Boolean Algebra. 6.9.3

Postulates of Boolean Algebra Postulate 1: (a) A = 0 if and only if A is not equal to 1 (b) A = 1 if and only if A is not equal to 0 Postulate 2: (a) x + 0 = x (b) x . 1 = x Postulate 3: Commutative Law (a) x + y = y + x (b) x • y = y • x Postulate 4: Associative Law (a) x + (y + z) = (x + y) + z (b) x . (y • z) = (x • y) • z Postulate 5: Distributive Law (a) x • (y + z) = x • y + x • z (b) x + y . z = (x + y) • (x + z) Postulate 6: (a) x + x = 1 (b) x • x = 0 The listed above are the basic axioms of the algebraic structure and need no proof. They are used to prove the theorems of Boolean algebra.

Minimization of Boolean Functions When a Boolean function is implemented with logic gates, each literal in the function designates an input to a gate, and each termis implemented with a gate. Hence for a given function, the minimization of the number of literals, and the number of terms, will result in a circuit with fewer components. 6.9.4

Minimization of Boolean Functions When a Boolean function is implemented with logic gates (discussed later in this chapter), each literal in the function designates an input to a gate and each term is implemented with a gate. Thus for a given Boolean function, the minimization of the number of literals and the number of terms will result in a circuit with less components.

A.B' + A.B = A.(B' + B) = A.T (X + X' = 1) = A (v) Simplify Y = A.B.C + A.B.C' + A.B' = A.B.(C + C') + A.B' = A.B + A.B' (X + X' = 1) = A.(B + B') = A (

A + B) + C = A + (B + (A • B) • A • (B • C) C) A(B + C) = A-C A + B ∎ C = (A + B) • (A + C) A+A-B=A A- (A + B) = A 10 A+ A -B = B A-(

| 123/158 | SUBMITTED TEXT | 23 WORDS | 95% | MATCHING TEXT | 23 WORDS |

computer by electronic/electrical components, such as transistors, semiconductors etc., all of which can only indicate two states or conditions - on(1) or

computer by electronic/electrical components such as transistors, semiconductors, wires, etc. all of which can only indicate two states or conditions - on(l) or 6

W https://pdfcookie.com/documents/36746371-foundations-of-computing-pradeep-k-sinha-bpb-publication ...

| 124/158 | SUBMITTED TEXT | 37 WORDS | 91% | MATCHING TEXT | 37 WORDS |

represented within the computer by the presence or absence of these types of signals. The binary number system, which has only two digits (0 and 1), is most suitable for expressing the two possible states. The

represented within the computer by the presence or absence of these various signals. The binary number system, which has only two digits (0 and 1), is most suitable for expressing the two possible states. The

W https://pdfcookie.com/documents/36746371-foundations-of-computing-pradeep-k-sinha-bpb-publication ...

| 125/158 | SUBMITTED TEXT | 17 WORDS | 81% | MATCHING TEXT | 17 WORDS |

algebra, which deals with the binary number system. It is very useful in designing logic circuits,

algebra that deals with the binary number system, is very useful in designing logic circuits

W https://pdfcookie.com/documents/36746371-foundations-of-computing-pradeep-k-sinha-bpb-publication ...

| 126/158 | SUBMITTED TEXT | 26 WORDS | 81% | MATCHING TEXT | 26 WORDS |

the processors of computer systems. Boolean algebra provides an economical and straightforward approach to the design of relay, and other types of switching circuits.

the design of modern computers. Boolean algebra provides an economical and straightforward approach to the design of relay and other types of switching circuits.

W https://pdfcookie.com/documents/36746371-foundations-of-computing-pradeep-k-sinha-bpb-publication ...

| 127/158 | SUBMITTED TEXT | 23 WORDS | 97% | MATCHING TEXT | 23 WORDS |

number system. 3. What is the difference between positional and non-positional number systems? Give examples of both types of number systems. 4.

number system. Questions 1. What is the difference between positional and non-positional number systems? Give examples of both types of number systems. 2.

W https://pdfcookie.com/documents/36746371-foundations-of-computing-pradeep-k-sinha-bpb-publication ...

| 128/158 | SUBMITTED TEXT | 19 WORDS | 84% | MATCHING TEXT | 19 WORDS |

Find the decimal equivalent of the following binary numbers: a. 101100001.1011 b. 11011101 c. 1011.1011 d. 1101011 5.

Find out the decimal equivalent of the following binary numbers: (a) 1101011 (f) 1000 (b) 11010 (g) 10110001100 (c) 10110011 (h) 1010101100 (d) 11011101 (

W https://pdfcookie.com/documents/36746371-foundations-of-computing-pradeep-k-sinha-bpb-publication ...

p 227 12 11 7 272 113 161 71 q 230 12 11 8 273 114 162 72 r 231 12 11 9 274 115 163 73 s 242 11 0 2 275 116 164 74 t 243 11 0 3 276 117 165 75 u 244 11 0 4 277 118 166 76 v 245 11 0 5 300 119 167 77 w 246 11 0 6 301 120 170 78 x 247 11 0 7 302 121 171 79 y 250 11 O 8 ' 140 122 172 7A z 251 11 0 9 : 072 123 173 7B { 300 12 0 # 043 124 174 7C | 152 12 11 @ 100 125 175 7D } 320 11 0 ' (apos) 047 126 176 7E - 241 11 0 1 = 075 127 177 7F DEL 007 12 7 9 " 042 128 200 80 040 11 0 1 303 8 9 129 201 81 041 0 1 9 a 141 130 202 82 042 0 2 9 b 142 131 203 83 043 0 3 9 c 143 132 204 84 044 0 4 9 d 144 133 205 85 025 11 5 9 e 145 134 206 86 006 12 6 9 f 146 135 207 87 027 11 7 9 g 147 136 210 88 050 0 8 9 h 150 137 211 89 051 0 1 8 9 i 151 138 212 8A 052 0 2 8 9 304 139 213 8B 053 0 3 8 9 305 140 214 8C 054 0 4 8 9 306 141 215 8D 011 12 1 8 9 307 142 216 8E 012 12 2 8 9 310 143 217 8F 033 11 3 8 9 311 144 220 90 060 12 11 0 1 8 9 312 145 221 91 061 1 9 j 152 146 222 92 032 11 2 89 k 153 147 223 93 063 3 9 l 154 148 224 94 064 4 9 m 155 149 225 95 065 5 9 n 156 150 226 96 066 6 9 o 157 151 227 97 ∅ 010 12 8 9

P Q R S T U V W X Y Z 0 1 2 3 4 5 6 7 8 9 101 101 101 101 101 101 101 101 101 101 101 101 101 0101 0101 0101 0101 0101 0101 0101 0101 0101 0000 0001 - 0010 0011 0100 0101 0110 0111 1000 1001 1010 0000 0001 0010 0011 0100 0101 0110 0111 1000 1001 50 51 52 53 54 55 56 57 58 59 5A 50 51 52 53 54 55 56 57 58 59 B C D E F G 1010 1010 1010 1010 1010 1010 1010 1010 0001 0010 0011 0100 0101 0110 0111 A2 A3 A4 A5 A6 A7 H I J K L M N O 1010 1010 1010 1010 1010 1010 1010 1010 1000 1001 1010 1011 1100 1101 1110

A 072 2 8 9 313 155 233 9B 073 3 8 9 314 156 234 9C 004 12 4 9 315 157 235 9D 024 11 4 9 316 158 236 9E 076 6 8 9 317 159 237 9F 341 11 0 1 9 320 160 240 A0 ∅ 101 12 0 1 9 321 161 241 A1 102 12 0 2 9 ~ 176 162 242 A2 103 12 0 3 9 s 163 163 243 A3 104 12 04 9 t 164 164 244 A4 105 12 0 5 9 u 165 165 245 A5 106 12 0 6 9 v 166 166 246 A6 107 12 0 79 w 167 167 247 A7 110 12 0 8 9 x 170 168 250 A8 111 12 1 8 y 171 169 251 A9 121 12 11 1 9 z 172 170 252 AA 122 12 11 2 9 322 171 253 AB 123 12 11 3 9 323 172 254 AC 124 12 11 4 9 324 173 255 AD 125 12 11 5 9 325 174 256 AE 126 12 11 6 9 326 175 257 AF 127 12 11 7 9 327 176 260 B0 130 12 11 8 9 330 177 261 B1 131 11 1 8 331 178 262 B2 142 11 0 2 9 332 179 263 B3 143 11 0 3 9 333 180 264 B4 144 11 0 4 9 334 181 265 B5 145 11 0 5 9 335 182 266 B6 146 11 0 6 9 336 183 267 B7 147 11 0 7 9 337 184 270 B8 ∅ 150 11 0 8 9 340 185 271 B9 151 0 1 8 341 186 272 BA 160 12 11 0 342 187 273

A B C D E F G 1010 1010 1010 1010 1010 1010 1010 0001 0010 0011 0100 0101 0110 0111 Al A2 A4 A5 A7 H I J K L M N O 1010 1010 1010 1010 1010 1010 1010 1010 1000 1001 1010 1011 1100 1101 1110 mi A8 A9 AA AB AC AD AE AF P 1011 0000 B0 Q 1011 0001 Bl R 1011 0010 B2 S 1011 0011 B3 T 1011 0100 B4 U 1011 0101 B5 V 1011 0110 B6 W 1011 0111 B7 X 1011 1000 B8 Y 1011 1001 B9 Z 1011 1010 BA

a layer of software on top of hardware, to manage all parts of the

a layer of software on top of the bare hardware, to manage all parts of the

| 136/158 | **SUBMITTED TEXT** | 16 WORDS | **76%** | **MATCHING TEXT** | 16 WORDS |

present the user an extended machine or virtual machine that is easier to program and

present the user with an interface or virtual machine that is easier to program and

W   https://pdfcookie.com/documents/36746371-foundations-of-computing-pradeep-k-sinha-bpb-publication …

| 137/158 | **SUBMITTED TEXT** | 22 WORDS | **73%** | **MATCHING TEXT** | 22 WORDS |

Process Management A process is a program in execution, which needs certain resources, including CPU time, memory, files, and I/O devices,

Process Management. A process is a program in execution. During execution, a needs certain resources such as CPU time, memory space, files, and I/O devices.

W   https://pdfcookie.com/documents/36746371-foundations-of-computing-pradeep-k-sinha-bpb-publication …

| 138/158 | **SUBMITTED TEXT** | 11 WORDS | **100%** | **MATCHING TEXT** | 11 WORDS |

File Management A file is a collection of related information

FILE MANAGEMENT A file is a collection of related information.

W   https://pdfcookie.com/documents/36746371-foundations-of-computing-pradeep-k-sinha-bpb-publication …

| 139/158 | **SUBMITTED TEXT** | 14 WORDS | **76%** | **MATCHING TEXT** | 14 WORDS |

It indicates the address of the next instruction to be executed for this

It holds the address of the next instruction to be executed. This

W   https://pdfcookie.com/documents/36746371-foundations-of-computing-pradeep-k-sinha-bpb-publication …

| 140/158 | **SUBMITTED TEXT** | 21 WORDS | **68%** | **MATCHING TEXT** | 21 WORDS |

Turnaround Time: - The interval from the time of submission of a process to the time of completion is the

Turnaround time the interval from the time of submission of a job to the system for processing to the time of completion of the

W   https://pdfcookie.com/documents/36746371-foundations-of-computing-pradeep-k-sinha-bpb-publication …

| 141/158 | **SUBMITTED TEXT** | 22 WORDS | **76%** | **MATCHING TEXT** | 22 WORDS |

Keep track of which parts of memory are in use. ➢ Allocate memory to processes when they need it. ➢ Deallocate

keep track of which parts of memory are in use and which parts are not in use, to allocate memory to processes when they need it and deallocate

W   https://pdfcookie.com/documents/36746371-foundations-of-computing-pradeep-k-sinha-bpb-publication …

142/158	SUBMITTED TEXT	20 WORDS	50% MATCHING TEXT	20 WORDS

into fixed sized blocks called Page frames. Logical memory address space/process is also broken into blocks of same size

into fixed-sized blocks called page frames. The total memory requirement of a process (including its instructions and data) is also into blocks of the same size

W   https://pdfcookie.com/documents/36746371-foundations-of-computing-pradeep-k-sinha-bpb-publication ...

143/158	SUBMITTED TEXT	15 WORDS	100% MATCHING TEXT	15 WORDS

to define the physical memory address that is sent to the memory unit.

to define the physical memory address that is sent to the memory unit.

W   https://pdfcookie.com/documents/36746371-foundations-of-computing-pradeep-k-sinha-bpb-publication ...

144/158	SUBMITTED TEXT	20 WORDS	100% MATCHING TEXT	20 WORDS

Page table is kept in main memory and a page table base register (PTBR) points to the page table.

SA   Operating System-edit.pdf (D143421170)

145/158	SUBMITTED TEXT	40 WORDS	57% MATCHING TEXT	40 WORDS

with this approach is much time require to access a user memory location. If we want to access some location i, we must get index into the page table from the PTBR offset. This task requires a memory access.

SA   OS_Book.pdf (D137278070)

146/158	SUBMITTED TEXT	23 WORDS	78% MATCHING TEXT	23 WORDS

with the page offset to produce the actual address. With this scheme two memory accesses are needed to access a byte.

SA   OS_Book.pdf (D137278070)

147/158	SUBMITTED TEXT	43 WORDS	89% MATCHING TEXT	43 WORDS

solution to this problem is to use a special small fast lookup hardware cache called Translation Look-aside Buffer (TLB). The TLB is associative high speed memory. Each entry in the TLB consists of two parts a key and a value. If the

SA   Operating System-edit.pdf (D143421170)

| 148/158 | SUBMITTED TEXT | 14 WORDS | 80% | MATCHING TEXT | 14 WORDS |

the page table is used to get the frame number from the page

The page table is used to obtain the corresponding page frame from the page

W https://pdfcookie.com/documents/36746371-foundations-of-computing-pradeep-k-sinha-bpb-publication ...

| 149/158 | SUBMITTED TEXT | 31 WORDS | 67% | MATCHING TEXT | 31 WORDS |

item is found the corresponding value field is returned. TLB search is fast but hardware is expensive. Generally the number f entries in TLB are small.

SA OS_Book.pdf (D137278070)

| 150/158 | SUBMITTED TEXT | 70 WORDS | 76% | MATCHING TEXT | 70 WORDS |

TLB contains only few of the page table entries. When a logical address is generated by the CPU, its page number is given to the TLB. If the page number is found its frame number is immediately available and used to access required memory. If the page number is not in the TLB (TLB miss) a memory reference to the page table is used to get the frame number

SA BALU-CSE-Operating System Course Material.pdf (D115578367)

| 151/158 | SUBMITTED TEXT | 15 WORDS | 84% | MATCHING TEXT | 15 WORDS |

are wired down. Some TLBs stores address space identifiers (ASIDs) in each entry

SA Operating System-edit.pdf (D143421170)

| 152/158 | SUBMITTED TEXT | 21 WORDS | 97% | MATCHING TEXT | 21 WORDS |

TLB. An ASID uniquely identifies each process and is used to provide address space protection for that process.

SA OS_Book.pdf (D137278070)

| 153/158 | SUBMITTED TEXT | 20 WORDS | 92% | MATCHING TEXT | 20 WORDS |

The percentage of time that a particular page number is found in the TLB is called the hit ratio.

SA BALU-CSE-Operating System Course Material.pdf (D115578367)