

# UNIT-I : COMPUTER SYSTEMS AND ORGANISATION

## CHAPTER-1

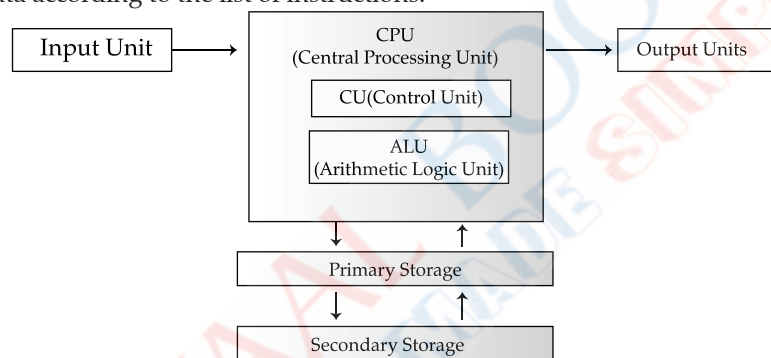
### BASIC COMPUTER ORGANISATION

#### Topic-1 Computer System



#### Revision Notes

- A computer is an electronic device that can perform a variety of operations in accordance with a set of instructions called program. In other words, we can say that computer is an electronic device which manipulates the data according to the list of instructions.



- **Computer System:** A computer system primarily comprises a central processing unit (CPU), memory, input/output devices and storage devices. All these components function together as a single unit to deliver the desired output. Below is the block diagram of Computer System.
- **Strengths of Computer:**
  - A computer is much faster as compared to human beings.
  - Computers are more accurate than a human for solving any problem.
  - They are immune to tiredness and boredom or fatigue.
  - Computers can perform repetitive jobs efficiently.
- **Weaknesses of Computer:**
  - Computer cannot take decisions on its own.
  - It needs to be told at each and every step to perform any certain tasks.
  - The computer don't have human-like feelings.
- A computer works on the IPO principle i.e.  
Input -> Process -> Output.
- A computer receives data as input, processes it, stores it, and produces output based on this, computer uses some components to take input, some components to do processing and some components to provide output.
- **Input Device:** The devices through which control signals are sent to a computer are termed as input devices. These devices convert the input data into a digital form that is acceptable by the computer system. Some examples of input devices include keyboard, mouse, scanner, touch screen, etc.
- **Output Device:** The device that receives data from a computer system for display, physical production, etc., is called output device. It converts digital information into human understandable form. For example, monitor, projector, headphone, speaker, printer, etc.
- **Central Processing Unit (CPU):** It is known as the brain of the computer. It has three main components:
  - **Control Unit (CU):** It controls and guides interpretation, flow and manipulation of data and information. It sends control signals until the required operations are properly done by ALU and memory unit (storage unit).
  - **Arithmetic and Logic Unit (ALU):** It performs all the required arithmetic and logical operations to perform the task.
  - **Memory Unit (Storage Unit):** It comprises of primary and secondary memory units.
- **Hardware:** All the physical components of the computer system are known as hardware e.g., keyboard, mouse, monitor, printer, etc.

- **Software:** Software represents the set of programs that govern the operations of a computer system and make the hardware run smoothly. Software can be classified into three types:
  - **System Software:** These softwares are mandatory for all computer systems to work. For example: Operating systems like MS-DOS, Microsoft Windows and Language processors like Assembler etc.
  - **Application Software:** These softwares are made to perform a specific task, For example: WordPad, MS-Word, etc.
  - **Utility Software:** Software that assists the computer to perform functions like data backup, virus removal, scanning, defragmenting, etc. For example: Antivirus, Disk defragmenter, etc.



## Key Terms

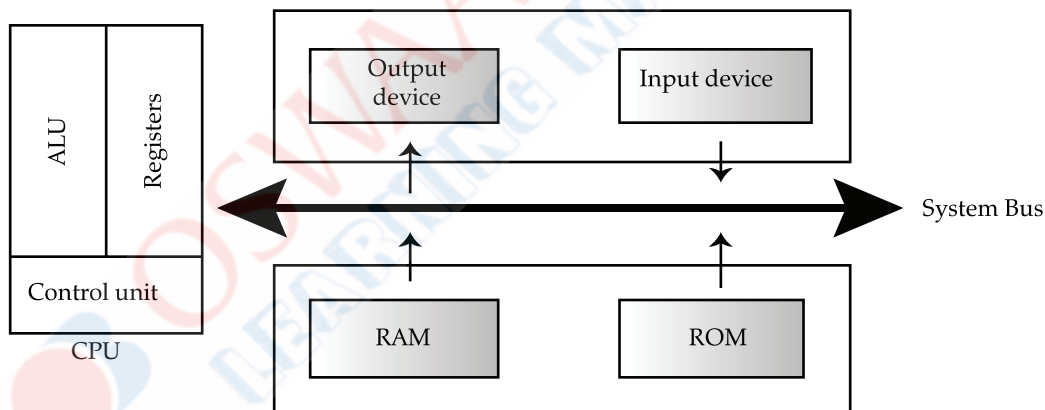
- **Data:** Data is the collection of facts and figures.
- **Information:** A meaningful content extracted from data.
- **Program:** A set of instructions that direct the computer.
- **Commands:** Special codes or keywords that the user inputs to perform a task.
- **Input:** Process of entering data and instructions in the computer.
- **Storage:** Saving data and instructions to make them readily available for future use.
- **Process:** Performing operations (arithmetic or logical) on data to convert them into useful information.
- **Output:** The process of producing useful information or result for the user.

## Topic-2 Central Processing Unit (CPU)



## Revision Notes

- CPU is a multipurpose, programmable logic device that reads binary instructions from a storage device called memory, accepts binary data as input and processes data according to those instructions and provides results as output. It is also referred to as a processor, central processor, or microprocessor.



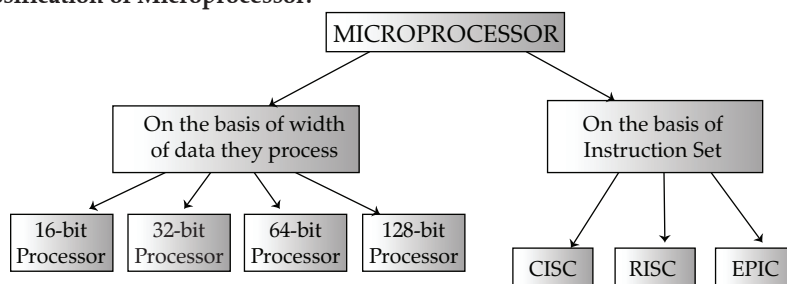
Block diagram of basic computer organisation

- **Components of Microprocessor:**
  1. CPU
  2. Bus
  3. Registers
  2. CU
- **CPU:** It is an acronym for Central Processing Unit. It controls all the operations of a computer. It has three components:
  - (i) Arithmetic Logic Unit (ALU)
  - (ii) Control Unit (CU)
  - (iii) Registers
- **Bus:** It refers to the collection of wires through which data is transmitted from one part of the computer to another. It is of 2 types: (i) Internal Bus, and (ii) External Bus.
- **Internal Bus:** It is a bus that connects all the internal components of the computer like CPU and main memory.
- **External Bus:** It connects the different external devices, peripherals, expansion slots, input/output ports and drive connections to the rest of the computer. External bus is also known as expansion bus.
- **Registers:** It is a small amount of very fast memory that is built into the CPU. It is very expensive.

► **Characteristics of microprocessor:**

- (i) Instruction Set
- (ii) Bandwidth
- (iii) Clock Speed

► **Classification of Microprocessor:**



## Key Terms

- **Instruction Set:** The set of instructions that a microprocessor can execute.
- **Band width:** The number of bits processed in a single instruction.
- **Clock speed:** The clock speed determines how many instructions per second a processor can process. It is given in megahertz or gigahertz.
- **CISC:** It is an acronym for Complex Instruction Set Computing. The CISC architecture contains a large set of computer instructions that range from very simple to very complex and specialised.
- **RISC:** It is an acronym for Reduced Instruction Set Computing. It has a relatively limited number of instructions. It is designed to perform a relatively small number of operations so that it can operate at a higher speed.
- **EPIC:** It is an acronym for Explicitly Parallel Instruction Computing. It refers to architecture in which features are provided to facilitate compiler enhancements of Instruction Level Parallelism in all programs while keeping hardware complexity relatively low.

## Topic-3 Memory Units

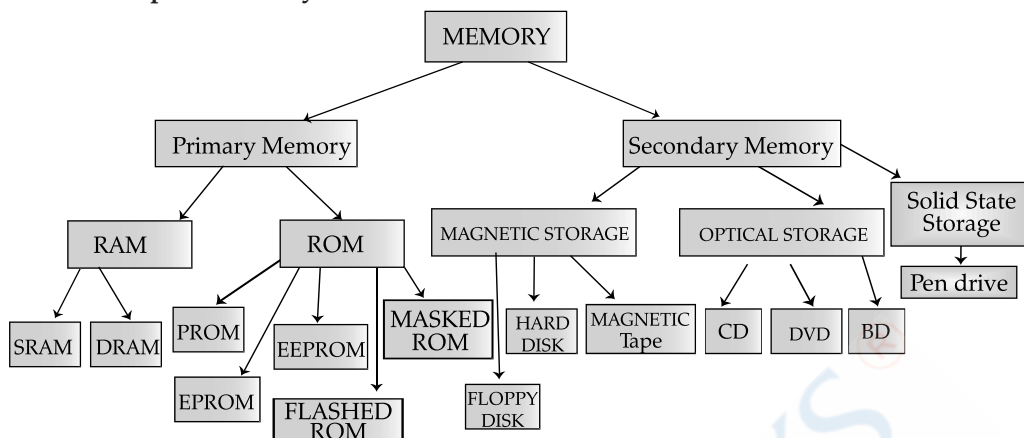


## Revision Notes

► **Order of Magnitude of Data:**

Metric system value (used to describe data transfer rate) (in Bytes)	Notation	Actual value (used to describe data storage) (in Bytes)
$10^3$	Kilo bytes (KB)	1024
$10^6$	Mega byte (MB)	$(1024)^2$
$10^9$	Giga byte (GB)	$(1024)^3$
$10^{12}$	Tera byte (TB)	$(1024)^4$
$10^{15}$	Peta byte (PB)	$(1024)^5$
$10^{18}$	Exa byte (EB)	$(1024)^6$
$10^{21}$	Zetta byte (ZB)	$(1024)^7$
$10^{24}$	Yotta byte (YB)	$(1024)^8$

► **Classification of Computer Memory:**



► **Primary Memory:** It is the internal storage that is used by our computer system to store data and instructions while processing. It is temporary memory and is volatile. It is directly accessed by the CPU. It is of two types viz. RAM and ROM.

1. **RAM:** It is an acronym for Random Access Memory. It is a type of Primary memory and is also known as read and write memory as it can be used for both purposes. It can be of the following two types:

(i) **SRAM:** It is an acronym for Static Random Access Memory. It is a type of memory that requires a constant power supply to hold the data and information. These are made of flip-flops.

(ii) **DRAM:** It is an acronym for Dynamic Random Access Memory. The term dynamic indicates that the memory is constantly needed to be refreshed otherwise it loses its contents. It is the most common kind of RAM used in personal computers. These are made of transistors and capacitors.

2. **ROM:** It is an acronym for Read Only Memory. It provides non-volatile storage of data. We can access any memory location of ROM by supplying its address. It is of the following types:

(i) **PROM:** It is an acronym for Programmable Read-Only Memory. It is a computer memory chip that can be programmed once it is created. But once it is programmed, then it cannot be changed or it is not re-programmable.

(ii) **EPROM:** It is an acronym for Erasable Programmable Read-Only Memory. It is a computer memory chip on which the written information can be changed as per the requirement. Its contents can be erased by using Ultraviolet light rays of a specific frequency for a specific period of time.

(iii) **EEPROM:** It is an acronym for Electrically Erasable Programmable Read-Only Memory. It is a PROM that can be erased and reprogrammed using an electric charge. In EEPROM, any location can be selectively erased.

(iv) **FLASHED ROM:** It is the same as EEPROM but is much faster. It erases but selective erasing is not possible.

(v) **MASKED ROM:** Its contents are programmed by the integrated circuit manufacturer (rather than by the user). The terminology mask comes from integrated circuit fabrication, where regions of the chip are masked off during the process of photolithography.

► **Secondary Memory:** It is also known as auxiliary storage. Unlike primary memory, it is used by our computer system to store a large amount of data. It is non-volatile and is not directly accessed by the CPU.

► **Magnetic Storage:** It is a magnetic medium that can be used to store a large volume of data. It uses a magnetic bead to read and write data. It gets the data to and from a magnetisable medium like plastic tape coated with fine particles of metal. It can store a large volume of data but this data is slowly accessed.

► **Optical Storage:** Optical storage medium holds data in digital form. It uses a laser to read and write data. Its storage capacity ranges up to 8GB.

► **Compact Disc (CD):** It is an optical media that is used for electronically recording, storing and playing back audio, video, text and other information in digital form.

► **Digital Video Disc (DVD):** It is also known as Super Density Disc (SDD) or Digital Versatile Disc (DVD). In appearance, they are somewhat similar to CDs but they have comparatively more storage capacities than that of CD.

► **Pen drive:** It is a storage device that includes flash memory with an integrated Universal Serial Bus (USB) interface. These are typically removable and rewritable and physically much smaller than an optical disc.

► **Blu-ray Disc (BD):** It is a digital optical disc data storage format designed to supersede the DVD format. It is capable of storing high-definition video resolution (1080p). It uses a blue-violet laser to read and writes data with high precision. Its storage capacity is up to 50 GB.

► **Cache Memory:** It is a smaller, faster memory used by a CPU to reduce the average time to access data from main memory.

## Topic-4 Input – Output (I/O) Devices



### Revision Notes

#### ➤ Input Devices

- Input unit is used to accept data and instructions from the user.
- Keyboard is used to type in letters, digits and commands.
- Mouse is a pointing input device.
- Microphone is used to send input to the computer.
- A scanner creates an electronic form of the printed image. Scanner are of three type: Handheld, Flatbed and Drum.
- A touch screen can detect where on the screen surface you are touching.
- A barcode reader is a handheld device that is used to obtain data contained in a bar code.
- Light pen is a handheld electro optical pointing device which is used for making drawings, graphics and for menu selection.

#### ➤ Output Devices

- Output unit is responsible for producing the output in user-readable form.
- Monitor displays information in a way similar to that shown on a television screen.
- The picture on a monitor is made up of thousands of tiny coloured dots called pixels.
- CRT (Cathode Ray Tube) contains an electron gun at the back of the glass tube.
- Liquid Crystal is a material used to create each pixel on the screen in LCD.
- TFT (Thin Film Transistor) is the device within each pixel that sets the charge.
- PDP (Plasma Display Panels) are flat panel displays.
- Plasma technology utilises small cells containing electrically charged ionised gases.
- Printer prints information and data from the computer onto a paper.
- Printer are divided into two categories: impact printer and Non-impact printer.
- Speakers receive the sound in form of electric current from the sound card.
- The Plotter is a graphic output device used to create drawings on paper.

➤ **Port:** A port is a connection point of interface between a computer and internal or external devices. Some common types of ports are - Serial Port, USB Port, Bluetooth, PS/2 Port, Infrared Port, Firewire Port.

➤ **Serial Port:** It is also known as COM Ports (communication ports). These are used to connect devices such as mouse, modem, etc. In this type of ports, only 1 bit of information can travel at a time.

➤ **Parallel Port:** It is a type of port in which data are sent and received simultaneously over several parallel channels. It can transfer up to 8 bits of data at a time. It is used to connect peripheral devices such as printer.

➤ **USB Port:** It is an acronym for Universal Serial Bus Port. It comes under a serial port and is used for short distance digital data communication. It allows data transfer between devices with little electric power.

➤ **PS/2 Port:** It is a type of port developed by IBM for connecting a mouse or a keyboard to the computer.

➤ **Infrared Port:** It is also known as an IR port. It is a meta port that sends and receives infrared signals from other infrared-enabled devices.

➤ **Firewire Port:** It is a serial bus interface standard for high-speed communication and real-time data transfer.

## CHAPTER-2

# SOFTWARE AND OPERATING SYSTEM

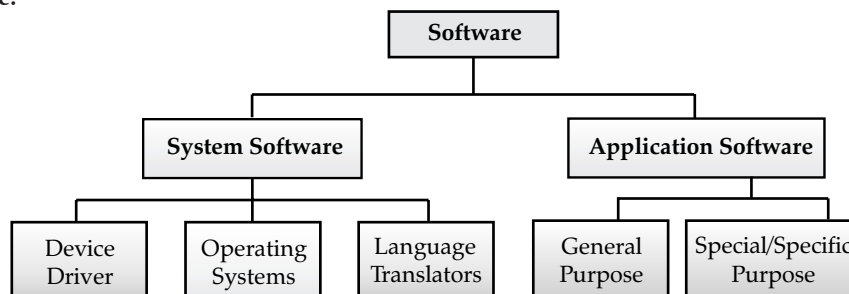
## Topic-1 Software Concept



### Revision Notes

➤ Software represents the set of programs that governs the operations of a computer system and make the hardware run smoothly.



► **Types of Software:**

- (i) **System Software:** These softwares are mandatory for all computer systems to work. For example: Operating Systems like Windows and language processors like assembler, etc.
- (ii) **Application Software:** These softwares are made to perform a specific task. For example: WordPad, MS-Word, etc.

**SYSTEM SOFTWARE**

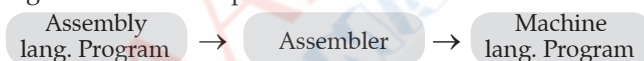
- (i) **Operating System:** An operating system(OS) is a system program that acts as an interface between user and the machine. It allocates and manages the resources of the computer.
- (ii) **Language Translators:** These programs are used to convert a program written in HLL(High Level Language) or assembly language to machine language.
- (a) **Compiler:** A compiler is a system program which translates a program written in a high level language into its equivalent program in machine language.



- (b) **Interpreter:** It converts a HLL program into equivalent machine language program one line at a time and executing it, if no error is encountered.



- (c) **Assembler:** It can be defined as a system program that converts an assembly language program into the machine language of the host computer.



- (iii) **Device Drivers:** A device driver is a program that controls a particular type of device that is attached externally to a computer.

► **UTILITY SOFTWARE**

These softwares are used to perform some additional functions which makes our computer more safe, secure and work smoothly. These softwares are considered to be a part of system software. For example, antivirus, disk defragmenter, etc.

- (i) **Antivirus:** An antivirus is utility software which detects and removes computer viruses. The antivirus keeps a watch on the functioning of the computer system.
- (ii) **Disk Defragmenter:** A disk defragmenter scans the hard disk for fragmented files and brings all the fragments together.
- (iii) **Backup Utility:** This utility is used to create the copy of the complete or partial data stored in a disk or CD on any other disk. In case the hard disk crashes or some other system failure occurs, the files can be restored using backup software.
- (iv) **Compression Utility:** This utility is used to compress large files. Compression is useful because it helps reduce resources usage and the file transmission on the network becomes easier.
- (v) **Disk Cleaner:** This utility scans for files that have not been accessed/used since long. Such files might be occupying huge amount of memory space. In that case, the Disk Cleaner utility prompts the user to delete such files so as to create more space on the disk.
- (vi) **File Management Tools:** This utility helps the user in storing, indexing, searching and sorting files and folders on the system. The most commonly used tool is the Windows Explorer and Google Desktop.

**APPLICATION SOFTWARE**

- (i) **General Purpose Application Software:** Some of the application software is made for the common users for day to day application and uses. The users may use them in the manner they want.
- **Word Processor:** Word processor is a general purpose application software used to create documents. It allows us to create, edit and format documents. Popular examples of Word processing software are Writer (Open Office) and Microsoft Word.
  - **Presentation Tools:** Presentation tools is a general purpose application software that lets us create presentations on any topic. We cannot only create a presentation and add slides but also can use different types of background, fonts, animations, audio, video, etc. We can add clip art and other graphics into our document. Even audio, video files can be added to the presentations. Popular examples of Presentation tool softwares are Impress (Open Office) and Microsoft Power Point.

- **Spreadsheet Packages:** Spreadsheet is a general purpose application software that lets us create and store data in tabular form. Both text and numerical values can be entered in a spreadsheet. All common mathematical and statistical formulae can be used on the numeric data. Popular examples of Spreadsheet software are Calc (Open Office) and Microsoft Excel.
  - **Database Management System:** Database Management System is general purpose application software that lets us create computer programs that control the creation, maintenance, and the use of database for an organisation and its end users. We cannot only store data but can also manage data in a database. Popular examples of Database Management System are Base (Open Office) and Microsoft Access.
  - **Graphics Software:** It is the application software which can manipulate images.
- (ii) **Special or Specific-Purpose (Customised Software):** Special purpose software applications are those which have been developed for a user for task which is very specific to them.
- **Customised Software** is one which is made as per the user's requirement. Such type of software is customer specific. It is made keeping in mind the individual needs of the user and so are also referred to as Domain Specific Tools. Such software cannot be installed and used by any other user/customer since the requirements may differ.
  - **Inventory Management System and Purchasing System:** Inventory Management System is generally used in departmental stores or other organisations to keep the record of the stock of all the physical resources.  
For example, in a Computer store, it keeps record of the number of computers, printers, printing sheet, printer cartridge available. It also helps to place purchase orders, bills, invoices etc. Various reports as position of stock, sales made in a particular period, profit earned etc. can be generated.
  - **School Management System:** School Management System (sometimes called a School Information System or SIS) is a system that manages all the school's data in a single, integrated application. Having all of the information in a single system allows schools to more easily connect data together. For example, when viewing a student's record, the user can follow a link to the student's class, and from there a link to the student's teacher, and from there a link to the teacher's other classes, and so on.
  - **Payroll Management System:** Payroll Management System software is used by all modern organisations to keep track of employees of the organisation who receive wages or salary. All different payment amounts are calculated by the payroll software and the record is maintained. The software keeps track of personal records of employees viz. name, address, date of birth, qualification, date of joining, etc. It also keeps track of professional record viz. allowances, perks, income tax, insurance, etc. Different reports, pay slips, etc. can be generated through this software.
  - **Financial Accounting:** Financial Accounting System is used to prepare accounting information, maintain different accounts ledger, and account books. It also helps an organisation to make budget.
  - **Hotel Management:** Hotel Management software refers to management techniques used in the hotel sector. These can include hotel administration, accounts, billing, marketing, housekeeping, front office or front desk, food and beverage management, catering and maintenance. Even advance bookings can be made through this software. Customers can have a look at the hotel and the rooms before making bookings. At any point of time the room availability, tariff for each type of room and even booking status can be checked.
  - **Reservation System:** Reservation System software is used to book (reserve) air flights, railway seats, movie tickets, tables in a restaurant, etc. In the case of a booking system, the inputs are booking requests. The processing involves checking if bookings are possible, and if so making the bookings. The outputs are booking confirmations/rejections.
  - **Weather Forecasting System:** This software makes it possible to forecast the weather for days and even months in advance. The detailed weather reports can also be generated.
- A **programming tool** or software development tool is a computer program that software developers use to create, debug, maintain, or otherwise support other programs and applications.
  - Typically, they are standalone utilities that provide or support a particular task within any phase of the development/programming cycle. For example, a debugger is a programming tool that helps programmers identify and resolve bugs within a program's source code.
  - Compilers, linkers, assemblers, disassemblers, load testers, performance analysts, GUI development tools and code editors are also all forms of programming tools.
  - It is important to understand here that computers and humans understand completely different languages. While humans are able to write programs in high-level language (HLL), computers understand machine language.
  - **Source code** refers to the original code written in a programming language by a programmer.
  - Machine code refers to the code converted into a form that a computer can understand and can directly execute.



## Key Terms

- Computer can deliver performance because of an efficient combination of Hardware and Software.
- **Device driver** acts as an interface between the device and the operating system.

## Topic-2 Operating System (OS)



## Revision Notes

- **Operating System** is defined as a collection of programs that coordinates the operations of computer hardware and software. It acts as a bridge or the interface between man and machine.
- Operating system is a system software which is mandatory for all computer systems to operate.
- **Need of Operating System**
- In earlier day's user had to design the application according to the internal structure of the hardware. Operating System was needed to enable the user to design the application without concerning the details of the computer's internal structure. In general the boundary between the hardware & software is transparent to the user.
- **Functions of an Operating System**
  - **Device management:** Operating system ensures the smooth functioning of all the peripheral devices.
  - **Processor management:** Operating system enables the activities of planning, performing and monitoring the performance of any process.
  - **Memory management:** It is the act of managing memory i.e., provide required memory for the process to happen and then save the desired result back in the memory.
  - **File management:** Every user of a computer system wants to manipulate some kind of information. This function of operating system allows the user to create a file, write into it, open it for reading, close it after reading or writing is over and delete it, if it is not required any more.
- **Types of Operating Systems**
  - **Real Time Operating System (RTOS):** It is a multi-tasking operating system that aims to execute real time applications. It is designed to respond to an event within a predetermined time. These are used to control machinery, scientific instruments and industrial systems.  
**For example:** Windows CE, Linux, etc.
  - **Multi user operating system :** It is the type of operating system that allows many users to take the advantage of computer's resources simultaneously.  
**For example:** UNIX, VMS, etc.
  - **Single user operating system:** As it is clear from the name, single user operating system is designed for one user to effectively use a computer at a time.  
**For example:** Windows 2007, Windows 10, etc.
  - **Single tasking operating system:** It is an operating system which can run only one program at a time. **For example:** Palm
  - **Multi-tasking operating system:** This operating system allows the execution of multiple tasks at a time.  
**For example:** Windows 2007/2008/2009 /XP, etc.
  - **Time sharing operating system:** It allows the users to share the computer resources simultaneously.  
**For example:** A mainframe computer that has many users logged on to it.
  - **Distributed operating system:** Distributed operating system uses multiple central processors to serve multiple real time applications.

### User Interface

- The feature of a computer system which allows the user to interact with it.
- The OS provides a user interface (UI), an environment for the user to interact with the machine
- A user interface, also sometimes called a human-computer interface, comprises both hardware and software components. It handles the interaction between the user and the system.
- There are different ways of interacting with computer systems which have evolved over the years.
- There are two prevalent types of user interface:
  - **Command Line Interface:** Command Line Interface requires users to type appropriate instructions into the command line. The computer is commanded to first go to the required file or directory. From there, a whole host of commands become available, from retrieving files to running programs.
  - **Graphical User Interface:** The graphical user interface, or GUI, is the type of interface with which the majority of people are most familiar. You interact with these interfaces by using a mouse, track pad, or other peripheral to point and click on graphics or icons.





## Key Terms

- ▶ **GUI:** Graphical User Interfaces
- ▶ **CLI:** Command Line Interfaces
- ▶ **POST:** Power On Self Test

# CHAPTER-3

## BOOLEAN ALGEBRA



## Revision Notes

- ▶ Boolean algebra developed by George Boole in 1854 deals with binary values and logical operations and hence is also known as **Binary Algebra** or **Logical Algebra**. It helps us in analysing and simplifying the digital circuits. It is also called Switching Algebra.
  - ▶ A **Boolean function** is defined by a Boolean expression consisting of binary variables, constants 0 or 1 and logical operators. For example: A Boolean function  $F(A, B, C)$  can be defined as:
 
$$F(A, B, C) = A.B + B.C + \bar{A} . \bar{C}$$
 Where LHS represents output Y of the logical expression. Hence, we can say
 
$$Y = A.B + B.C + \bar{A} . \bar{C}$$
  - ▶ A **Boolean expression** is a logical expression that produces a Boolean value *i.e.*, true or false when evaluated.
  - ▶ Boolean Algebra uses three basic logical operators:-
    - **NOT operator** – It operates on single input and gives complement of the input as output. It is represented by an over bar or apostrophe on the variable to be operated. Thus if  $A = 0$ , then  $A' = 1$  and vice versa. Most programming languages use a '!' (exclamation sign) as NOT operator.
    - **OR operator** – It is a binary operator equivalent to a logical (+) addition. Thus  $A + B + C$  implies **A OR B OR C**. This operator gives a true (*i.e.* 1) if any of the operand is True or 1.
    - **AND operator** – It is a binary operator equivalent to logical (·) multiplication. Thus  $A \cdot B$  implies **A AND B**. It gives a true (*i.e.* 1) only if both the operands are True or 1.
  - ▶ A logical variable can take **only** two values *i.e.* either a binary 1 or a binary 0. Although, there can be infinite number of variables in a Boolean expression. While evaluating boolean expression the order of evaluation of logical operators is NOT, AND then OR, Parenthesis is evaluated first.
  - ▶ **Truth Table** – It is tabulated form of all the possible input combinations and their respective outputs. The number of possible input combinations for a Boolean expression is  $2^n$ , where n is the number of input variables. Hence, for a Boolean function  $F(A, B)$ , number of possible combinations will be  $2^2$  *i.e.* 4 and for a Boolean function  $F(A, B, C, D)$  it will be  $2^4$  *i.e.* 16.
- For the function  $F(A, B, C) = A.B + B.C + \bar{A} . \bar{C}$ , there will be 8 possible combinations and hence 8 rows in the truth table as given below.

Example of Truth Table:

A	B	C	A.B	B.C	$\bar{A} . \bar{C}$	Y
0	0	0	0	0	1	1
0	0	1	0	0	0	0
0	1	0	0	0	1	1
0	1	1	0	1	0	1
1	0	0	0	0	0	0
1	0	1	0	0	0	0
1	1	0	1	0	0	1
1	1	1	1	1	0	1

► **Basic Rules (Postulates) of Boolean Algebra**

- $X + 0 = X$  or  $0 + X = X$
- $X + 1 = 1$  or  $1 + X = 1$
- $X \cdot 0 = 0$
- $X \cdot 1 = X$

► Laws used in Boolean Algebra are summed up in the table given below

LAW	AND	OR
Commutative Law	$A \cdot B = B \cdot A$	$A + B = B + A$
Associative Law	$A \cdot (B \cdot C) = (A \cdot B) \cdot C$	$A + (B + C) = (A + B) + C$
Distributive Law	$A \cdot (B + C) = A \cdot B + A \cdot C$	$A + (B \cdot C) = (A + B) \cdot (A + C)$
Identity Law	$A \cdot 1 = A$	$A + 0 = A$
Idempotent Law	$A \cdot A = A$	$A + A = A$
Complement Law	$A \cdot \bar{A} = 0$	$A + \bar{A} = 1$
Absorption Law	$A \cdot (A + B) = A$	$A + (A \cdot B) = A$
Double Negation or Involution Law	$A'' = A$	

► **De Morgan's theorems**

(i)  $\overline{X \cdot Y} = \bar{X} + \bar{Y}$       (ii)  $\overline{X + Y} = \bar{X} \cdot \bar{Y}$

- The dual principle or principle of duality says that the boolean algebra remains unchanged when the dual pairs are interchanged. But nothing goes with complement because complement is self dual operation.
- Principle of Duality states that starting with a boolean relation, another boolean relation can be derived by
1. Changing each OR sign (+) to an AND sign (·).
  2. Changing each AND sign (·) to an OR sign (+).
  3. Replacing each 0 by 1 and each 1 by 0.
- **Logic circuit**, electric circuit whose output depends upon the input in a way that can be expressed as a function in symbolic logic; it has one or more **binary inputs** (capable of assuming either of two states, e.g., On or Off) and a single **binary output**. Logic circuits that perform particular functions are called **logic gates**.
- Logic Gate is simply an electronic circuit which takes one or more electronic signals as input to give an output.
- **NOT Gate** is an inverter circuit that operates on single input. It gives complement of the input as the output. It is represented by the following symbol.



Truth table of NOT gate

A	Y = A'
0	1
1	0

- **AND Gate** gives a high output if both the inputs are high. Algebraically it is represented by a (·) dot. Its electronic symbol is



Truth table of AND gate

A	B	X = A.B
0	0	0
0	1	0
1	0	0
1	1	1

- **OR Gate** gives a high input if any one of the inputs is high. Its algebraic symbol is (+) Plus, and electronically it is represented as

**Truth table of OR gate**

A	B	$X = A + B$
0	0	0
0	1	1
1	0	1
1	1	1



- **NAND Gate** is inverter of AND gate. It gives a High Input if any of the inputs is low. Its symbol is AND Gate with a small circle at the output implying inversion.

**Truth table of NAND gate**

A	B	$X = \overline{A \cdot B}$
0	0	1
0	1	1
1	0	1
1	1	0

- **NOR Gate** is inverter of OR. Its output is low if any of the inputs is high. Its symbol is OR Gate with a small circle at the output implying inversion.

**Truth table of NOR gate**

A	B	$X = \overline{A + B}$
0	0	1
0	1	0
1	0	0
1	1	0



- **XOR Gate** gives a high output if odd number of inputs is high. It is used to detect the mismatch of bits. Its algebraic symbol is  $\oplus$ . Its electronic symbol is



$$Q = A \oplus B \text{ or } Q = \overline{A} \cdot B + A \cdot \overline{B}$$

**Truth table of XOR gate**

A	B	$Q = A \oplus B$
0	0	0
0	1	1
1	0	1
1	1	0

- **XNOR Gate** gives a high output when both the inputs are same i.e., either both are high or both are low. Its electronic symbol is

**Truth table of XNOR gate**

A	B	$X = A \odot B$
0	0	1
0	1	0
1	0	0
1	1	1



$$Y = A \odot B \text{ or } Y = \overline{A} \overline{B} + AB$$

- **Tautology:** It is an expression or assertion that is always true.
- **Fallacy:** An expression that always yields 0 or false is called Fallacy.
- **Universal gates** – NAND and NOR gates are known as universal gates as any possible circuit can be built using these two gates.

- The decision which results into either Yes (TRUE) or No (FALSE) is called a Binary Decision.
- Statements which can be determined to be true or false are called logical statements truth functions and the results TRUE or FALSE are called Truth values.



## Key Terms

- **AND – OR - Invert Logic:** When the output of an AND – OR circuit is complemented (inverted), it results in an AND – OR – Invert circuit.
- **AND – OR Logic:** AND - OR circuit can have any number of AND gates each with any number of inputs.
- **Exclusive – NOR Logic:** The exclusive – NOR can be implemented by simply inverting the output of an exclusive – OR (XNOR).
- **Exclusive – OR logic:** This circuit is considered a type of logic gate with its own unique symbol. It is actually a combination of two AND gates, one OR gate, and two inverters (NOT).
- **NAND Gate:** The NAND gate is a universal gate because it can be used to produce the NOT, the AND, the OR, and the NOR functions.
- **NOR Gate:** The NOR gate is a universal gate because it can be used to produce the NOT, AND, OR and NAND functions.

## CHAPTER-4

# NUMBER SYSTEM AND ENCODING SCHEMES

## Topic-1 Number System



## Revision Notes

- A set of values used to represent different quantities is known as number system.
- The four most common number systems are:
  - (1) **Decimal Number System:** The decimal number system has base or radix 10 because it uses ten digits from 0 to 9. In the decimal number system, the positions starting from the rightmost digit moving towards left represent units, tens, hundreds, thousands, and so on, respectively.
  - (2) **Binary Number System:** The Base-2 number system is also famously known as the Binary Number System wherein only two binary digits exist, i.e., 0 and 1. The base of binary number system is 2, because it has only two digits.
  - (3) **Octal Number System:** Octal number system has only eight (8) digits from 0 to 7. Every number (value) represents with 0,1,2,3,4,5,6 and 7 in this number system. The base of octal number system is 8, because it has only 8 digits.
  - (4) **Hexadecimal Number System:** A Hexadecimal number system has sixteen (16) alphanumeric values from 0 to 9 and A to F. Every number (value) represents with 0,1,2,3,4,5,6, 7,8,9,A,B,C,D,E and F in this number system. The base of hexadecimal number system is 16, because it uses 16 alphanumeric values. Here A is 10, B is 11, C is 12, D is 13, E is 14 and F is 15.
- Relation between various number systems is given below:

### Conversion between Number Systems

Decimal	Hexadecimal	Octal	Binary
0	0	0	0000
1	1	1	0001
2	2	2	0010
3	3	3	0011
4	4	4	0100
5	5	5	0101
6	6	6	0110
7	7	7	0111
8	8	10	1000
9	9	11	1001



10	A	12	1010
11	B	13	1011
12	C	14	1100
13	D	15	1101
14	E	16	1110
15	F	17	1111

### ► Binary Arithmetic

In binary number system, these are only 2 digits 0 and 1 and any number can be represented by these two digits. The arithmetic of binary number means the operation of addition, subtraction, multiplication and division. Binary arithmetic is essential part of all the digital computers and many other digital system.

- **Binary Addition:** In any number system, the concept of addition states that adding unit value (1) to any number results in the next number in the number line.

There are four rules of binary addition

A+B	Sum	Carry
0+0	0	1
0+1	1	0
1+0	1	0
1+1	0	1

In fourth case, a binary addition is creating a sum of  $(1 + 1 = 0)$  i.e., 0 is written in sum and carry 1 to the next column.

$$\begin{array}{r} \text{e.g.} \quad \quad \quad 11 \leftarrow \text{Carry} \\ \quad \quad \quad 11101 \\ \quad \quad \quad \underline{100} \\ \quad \quad 10001 \end{array}$$

- **Addition fractions:** Make both the number have equal number of digit by putting leading and following zeros.
- Start addition from the fractional part without disturbing the position of the point.

- **Binary subtraction:** There are three methods of performing binary subtraction:

(a) **Method of borrow:** In any number system, the concept of subtraction states that subtracting unit value. (1) From any number system the previous number in the number line.

There are four rules of Binary subtraction:

A-B	Subtract	Borrow
0-0	0	0
1-0	1	0
1-1	0	0
0-1	1	1

e.g.  $(1010)_2 - (111)_2$

$$\begin{array}{r} 1010 \\ \underline{111} \\ 0011 \end{array}$$

(b) **Method of one's complement:** One complement of a binary number is found by inverting the zero and one's without changing the number of digits.

The steps to be followed in subtraction by 1's complement are:

- To write down 1's complement of the subtrahend.
- To add this to the minuend.
- If there is no carry over, then 1's complement of the result of addition is obtained to get the final result and it is negative.

e.g.,  $101011 - 111001$

1's complement of 111001 is 000110

Minuend  $101011$

1'st complement  $000110$

$001101$

Here, there is no carry, answer is (1's complement of the sum obtained 11001) i.e., 001110.

(c) **Method of two's complement:** Two's complement of a binary number is found by adding 1 to its one's complement.

**Steps:**

- At first 2's complement of the subtrahend is found.
- Then it is added to the minuend.
- If the final carry over of sum is 1 it is dropped and result is positive
- If there is no carry over, the 2's complement of sum will be the result and it is negative.

e.g.  $10110 - 11010$

2's complement of 11010 is  $(00101 + 1)$

i.e., 00110

Hence        minuend                      1 0 1 1 0

2's complement of subtrahend  $+ 0 0 1 1 0$

1 1 1 0 0

As there is no carry over, the result of subtraction is negative and is obtained by writing the 2's complement of 11100 i.e.,  $(000110 + 1)$  or 00100

Hence the difference is  $-100$

- **Binary Multiplication:** Binary Multiplication is done, similarly the multiplication of decimal number. There four rules of Binary Multiplication.

A × B	Multiplication
$0 \times 0$	0
$0 \times 1$	0
$1 \times 0$	0
$1 \times 1$	1

e.g.  $11010 \times 10$

$$\begin{array}{r}
 11010 \\
 \times 10 \\
 \hline
 00000 \\
 11010 \times \\
 \hline
 1101001
 \end{array}$$

- **Binary Division:** It is similar to long division Binary division comprised of other two binary arithmetic operation as multiplication and subtraction.

$$\begin{array}{r}
 101 \overline{) 11010} \left( 101 \right. \\
 \underline{101} \phantom{00} \\
 110 \phantom{0} \\
 \underline{101} \phantom{0} \\
 1
 \end{array}$$



## Key Terms

- **Bit:** A single binary digit is abbreviated as bit.
- **Base or radix:** Number of distinct symbols that can be used to represent numbers in a number system is its base.
- **LSD (Least significant digit):** It is the digit which carries least weight (has the minimum value) in every number system.
- **MSD (Most Significant Digit):** In every number system the digit of a number that carries maximum weight (has the highest value) is the most significant Digit.
- **LSB (Least Significant Bit):** In a binary number, the light most bit carries the smallest weight and hence is called LSB.
- **MSB (Most Significant Bit):** In a binary number, the left bit carries the maximum weight and hence is called MSB.

## Topic-2 Encoding Schemes



### Revision Notes

- Integers may be represented in various ways in computers. These are
  - (a) Sign and magnitude representation
  - (b) One's complement, Two complement
- As we know that all digital computers store data consisting of digits, alphabets, symbols, etc. This data is not stored in the form we input it but it is stored in the form of binary code. Each alphabet or digit or symbol have their unique binary code.
- Some of the main computer codes are:
  - (i) **ASCII Code:** It is acronym for the American Standard Code for Information Interchange. It is used in most microcomputers and minicomputers and in many mainframes. It is a 7-bit code so it has  $2^7 = 128$  possible code groups. Extended ASCII uses 8 bits to represent a character so  $2^8 = 256$  different characters are possible.
  - (ii) **ISCII Code:** It is acronym for Indian Standard Code for Information Interchange. It is a 8-bit code so it has  $2^8 = 256$  possible code groups. It retains all ASCII characters and offers coding for Indian characters also.
  - (iii) **UNICODE:** It is the new universal coding standard being adopted by all newer platforms. Unicode provides a unique number for every character, no matter what the platform or program or the language is. Commonly used UNICODE encodings are UTF-8, UTF-16 and UTF-32. It is a superset of ASCII, and the values 0–128 have the same character as in ASCII.
- Character encoding is a way to convert text data into binary number.
- Encoding: It is a way to convert data from one format to another.
- In nutshell, We can assign unique numeric values to specific characters and convert those numbers in binary language.



### Key Terms

- **Encoding Scheme:** It refers to the way in which every character of the information is converted into machine understandable form.
- **Code Space:** It is the set of all the codes used by an encoding scheme to represent characters. It ranges from 0 to  $2^n - 1$  code where  $n$  is the number of bits used by that scheme.
- **Code Point:** It is the code for a single character; any scheme has total number of bits used to represent a single character in that particular encoding scheme.
- **Code Unit:** It is the maximum number of bits used to represent a character in any encoding scheme.

## UNIT-II : COMPUTATIONAL THINKING AND PROGRAMMING – 1

### CHAPTER-5


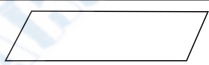

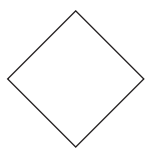
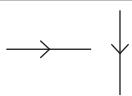
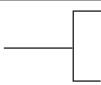
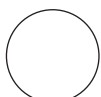
## INTRODUCTION TO PROBLEM SOLVING



### Revision Notes

- A software is developed in order to automate any daily life task that a human does.
- To automate any task, precise step by step instructions are required to be given.
- The success of any automated system depends on the precision of these instructions.
- Computerising any system requires a step by step approach.
- This step by step approach to problem-solving is known as a problem solving cycle.

- Following are the steps of this cycle.
  - **Problem Analysis** involves identifying the problem and the input data and the output desired.
  - **Developing an algorithm** is the step where all the possible solutions to the problem are identified and the best out of these is represented as an algorithm in a natural language.
  - **Coding** is done using any of the programming languages based on the algorithm developed.
  - **Testing and debugging** is required to check if the developed system produces the required output and any errors found are removed.
- In today's world of ever changing technology and user requirements, there is always a constant need for upgrading the system so all these steps are performed in cyclic order over the period.
- **Algorithm:** A step by step sequence of performing a task is called algorithm.
  - Algorithm is required by a programmer to clearly visualize the instructions. It is required in order to develop a software. These also help to identify the proper inputs and correct outputs.
  - A good algorithm must
    - Precisely describe the Steps
    - Define the results of each step uniquely and ensure that result of each step depends purely on the inputs of that step or the output of the preceding steps.
    - Stop after a finite number of steps.
    - Receive some input
    - Produce some output.
  - Following are considered while writing an algorithm
    - Input
    - Processing to be done
    - Output
  - Two most common methods to represent an algorithm are
    - Pseudo code
    - Flowchart
  - Both these methods should
    - Clearly depict the logic of the solution.
    - Show the flow of control during the program execution.
- **Flowchart**
  - A flowchart is a pictorial representation of step by step solution of a problem.
  - Symbols used in a flowchart are as follows:

Symbol	Purpose
	Start /Stop
	Input / Output
	Processing
	Decision box
	Flow lines
	Annotation
	Connectors

- **Pseudocode:** Algorithms can also be represented using pseudocode. Instructions of a pseudocode cannot be executed directly by the computer, but can be easily understood by the programmer.



- Pseudocode instructions are written using keywords of which some are given below
    - INPUT
    - PRINT
    - IF/ELSE
    - WHILE
    - TRUE/FALSE
    - COMPUTE/CALCULATE
    - DISPLAY
    - FOR
    - CASE
  - **Advantages of a pseudocode**
    - It improves the readability of an algorithm.
    - It can be used as a rough documentation for increased program understandability of programs by different programmers.
    - Coding becomes easy
- Flow of Control** is used to show the sequence or flow of events as will happen during program execution. This flow can be sequential or branch out based on a condition or even iterative.
- **Sequential flow:** It has events occurring in a sequence one after the other without being dependent on any condition.
  - **Selective flow:** Here the flow of control gets branched based on whether a particular condition evaluates to true or false.
  - **Iterative flow:** Here a sequence of steps is performed iteratively until some condition is met.
- **Verifying algorithm:** Any algorithm needs to be verified in terms of expected inputs and required outputs using dry run. In the due course the algorithm is verified and modified.
  - **Comparing algorithms:** Various approaches applied to solve a problem need to be compared to find out the best possible solution. For these algorithms are compared for their time and space complexity.
  - **Decomposition:** It is breaking a complex problem into multiple sub problems such that each sub problem can be examined and solved independently and combined later to arrive at the required system.

## CHAPTER-6

# BASICS OF PYTHON PROGRAMMING



### Revision Notes

- Python programming language was developed by Guido Van Rossum in Feb 1991. It was named after famous BBC comedy show Monty Python's Flying Circus.
  - It is an easy to learn yet powerful object oriented programming language. It is a very high level programming language.
  - Python is an interpreted language.
  - Python's interactive interpreter is also called Python shell.
  - **Variables** are reserved memory locations to store values. A variable has a name and a value.
  - Interactive mode of working means we type the command – one command at a time, and the Python executes the given command there and then gives us output.
  - Interactive mode is used for testing code.
  - Script mode is useful for creating programs and then run the programs later and get complete output.
  - **Comments:** A hash sign (#) that is not inside a string literal begins a single line comment. We can use triple quoted string for giving multiple-line comments.
  - **Character set:** It is a set of valid characters that a language can recognise. A character represents any letter, digit or any other symbol.
  - **Token:** The smallest individual unit in a program is known as a token or a lexical unit.
- There are different types of token in Python such as identifier, reserved words, literals, operators and punctuators.

- **Identifier:** An identifier is a name used to identify a variable, function, class, module or other object. An identifier starts with a letter A to Z or a to z or an underscore ( `_` ) followed by zero or more letters, underscores and digits (0 to 9). Python does not allow punctuation characters such as `@`, `&` and `%` within identifiers. Python is a case sensitive programming language.
  - **Reserved words (keywords):** Python has some reserved words. These reserved words may not be used as constant or variable or any other identifier names. All the python keywords contain lowercase letters only except False, None and True which have first letter capital.
  - **Literals/Constants:** Literals are data items that have a fixed value. Python allows several kind of literals as String literals, Numeric literals, Boolean literals, special literal 'None' and literal collections.
  - **Operator:** Operators are special symbols which perform some computation. Operators and operands form an expression.
  - **Punctuator:** These are symbols that are used in a programming language to organise programming sentence structures and indicate the rhythm and emphasis of expression, statements and program structure.
- **L-value and R-value:** Python uses the concept of L-value and R-value that is derived from the typical mode of evaluation on the left and right side of an assignment statement.
- L-value:** This concept refers to the requirement that the operand on the left side of the assignment operator is modifiable, usually a variable.
- R-value:** This concept pulls of fetches the values of the expression or operand on the right side of the assignment operator.
- **Input and Output**
- A program needs to interact with the user to accomplish the desired task, this done using for Input-Output facility.
- **Input( )** function first takes the input from the user and then evaluates the expression which means Python automatically identify whether user entered a string number or list.
- Syntax:**
- ```
Variable_name = input (<message to be displayed>)
```
- print( ):** To output your data to the screen, use the print() function. Print evaluates the expression before printing it on the monitor.
- Syntax:**
- ```
print (expression)
```

## CHAPTER-7

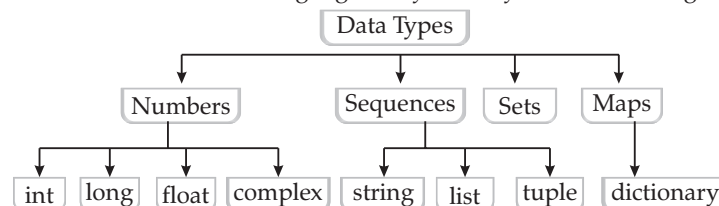
# DATA HANDLING

### Topic-1 Data Types, Operators & Expressions



#### Revision Notes

- Data type is a set of values and the allowable operations on those values. Python has a great set of useful data types. Python's data types are built in the core of the language. They are easy to use and straightforward.



- **Numbers** can be either integers or floating point numbers.
- Integers are whole numbers. There are three types of integers in Python: (a) int (Plain integers), (b) Long, (c) Booleans.
- (a) Plain Integers use 4 bytes to store a value these can be both positive and negative.

- (b) Long integers are larger than integers and can store a number as large as the available memory of the machine.
- (c) Boolean represents truth values. False and True and behave like the values 0 and 1.

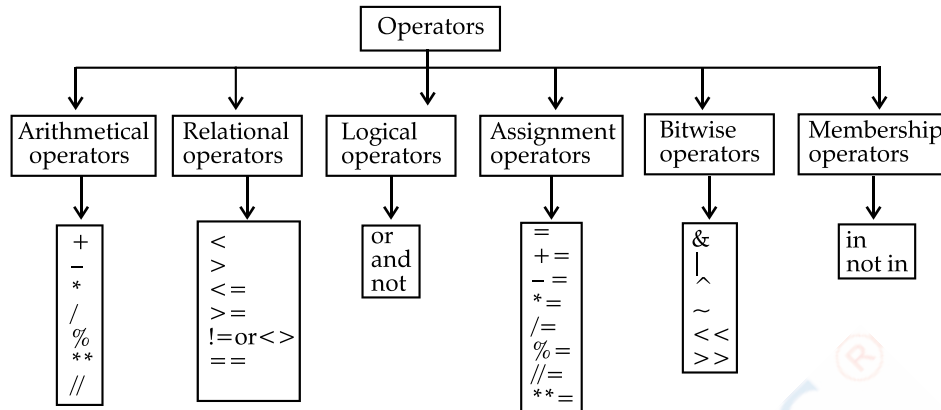
- Float or floating point numbers are numbers having a fractional part. They represent numbers with 15 digit precision.
- Complex numbers are represented as  $A+Bj$ , where A is the real part and B is the imaginary part and both are represented as floating point numbers.
- A **sequence** is an ordered collection of items, indexed by integers starting from 0. Sequences can be grouped into strings, tuples and lists.
  - **Strings** are single line text that can contain any characters. They can be declared with single or double quotes.
  - **Lists** are used to group other data. They are similar to arrays.
  - A **tuple** consists of a number of values separated by commas.
- A **set** is an unordered collection with no duplicate items.
- A **dictionary** is an unordered set of key:value pairs where the keys are unique.

#### Data type conversions:

There are several built-in functions to perform conversions from one type to another.

Function	Description
<code>int(x [,base])</code>	Converts x to an integer. base specifies the base if x is a string.
<code>long(x [,base] )</code>	Converts x to a long integer. base specifies the base if x is a string.
<code>float(x)</code>	Converts x to a floating-point number.
<code>complex(real [,imag])</code>	Creates a complex number.
<code>str(x)</code>	Converts object x to a string representation.
<code>repr(x)</code>	Returns a string containing a printable representation of an object.
<code>eval(str)</code>	Evaluates a string and returns an object.
<code>tuple(s)</code>	Converts s to a tuple.
<code>list(s)</code>	Converts s to a list.
<code>set(s)</code>	Converts s to a set.
<code>dict(d)</code>	Creates a dictionary. d must be a sequence of (key: value) pair.
<code>frozenset(s)</code>	Converts s to a frozen set.
<code>chr(x)</code>	Converts an integer to a character.
<code>unichr(x)</code>	Converts an integer to a Unicode character.
<code>ord(x)</code>	Converts a single unicode character to its integer unicode value.
<code>hex(x)</code>	Converts an integer to a hexadecimal string.
<code>oct(x)</code>	Converts an integer to an octal string.

- Mutable data types can change their values in place e.g. Lists and dictionaries.
- Three key attributes associated with any Python object are:
  - (a) The type that defines the operations that can be performed on an object. `type ()` returns the type of an object.
  - (b) The value of an object is the data item in the object. `print ()` statement is used to display the value of an object.
  - (c) The id of an object represents the location in memory of the object and does not change once it is created.
- An expression is a combination of operators, literals and variables.
- A compound expression contains multiple types of operators.
- An expression in Python can be one of the following types.
  - (a) Arithmetic expression that involves numbers and arithmetic operators.
  - (b) Logical expression has literal and/ or variables and logical operators.
  - (c) String expression contains string operators (+,\*) along with string operands and integers.
    - + with a string acts as concatenation operator for two strings.
    - \* with a string and one integer acts as a replication operator.
- In a mixed arithmetic expression, Python converts all operands type to the highest data type among the operands. This is called implicit type conversion.
- When a user explicitly converts data type of a variable it is called type casting.  
Syntax `<data type> (expression)`
- An **implicit** type conversion is a conversion performed by the compiler without programmer's intervention.
- An **explicit** type conversion (type promotion), is done by Python where it converts all the operands to the largest operand.  
Example: `a,b=3,5`  
`c= a/b` → c will always have floating point result.
- Operators are special symbols which perform some computation. Operators and operands form an expression.
- Python operators can be classified as given below.



### ➤ Arithmetical Operators

- Arithmetic operators help us to perform various arithmetic calculations. The arithmetic operators are explained in the table below. Let's assume value of  $a=2$  and  $b=3$

Operator	Description	Example
+	Addition - Adds values on either side of the operator.	$a + b$ results in 5
-	Subtraction - Subtracts right hand operand from left hand operand.	$a - b$ results in -1
*	Multiplication - Multiplies values on either side of the operator.	$a * b$ results in 6
/	Division - Divides left hand operand by right hand operand.	$a/b$ results in 0
%	Modulus - Divides left hand operand by right hand operand and returns remainder.	$a \% b$ results in 2
**	Exponent - Performs exponential (power) calculation on operators.	$a ** b$ results in 8
//	Floor Division - The division of operands where the result is the quotient in which the digits after the decimal point are removed.	$11//2$ results in 5 $11.0//2.0$ results in 5.0

### ➤ Relational/Comparison Operators

- These operators help us to make decisions based on certain conditions.

Operator	Description	Example
= =	Checks if the value of two operands are equal or not, if yes then condition becomes true.	$(a == b)$ is not true.
!=	Checks if the values of two operands are equal or not, if values are not equal then condition becomes true.	$(a != b)$ is true.
< >	Checks if the value of two operands are equal or not, if values are not equal then condition becomes true.	$(a < > b)$ is true. This is similar to != operator.
>	Checks if the value of left operand is greater than the value of right operand, if yes then condition becomes true.	$(a > b)$ is not true.
<	Checks if the value of left operand is less than the value of right operand, if yes then condition becomes true.	$(a < b)$ is true.
> =	Checks if the value of left operand is greater than or equal to the value of right operand, if yes then condition becomes true.	$(a > = b)$ is not true.
< =	Checks if the value of left operand is less than or equal to the value of right operand, if yes then condition becomes true.	$(a < = b)$ is true.

### ➤ Assignment Operators

- These operators assign the right side value to the left side variable.

Operator	Description	Example
=	Simple assignment operator. Assigns values from right side operands to left side operand.	$c = a + b$ will assign the value of $a + b$ into $c$
+=	Add AND assignment operator. It adds right operand to the left operand and assign the result to left operand.	$c += a$ is equivalent to $c = c + a$



Operator	Description	Example
-=	Subtract AND assignment operator. It subtracts right operand from the left operand and assign the result to left operand.	c -= a is equivalent to c = c - a
*=	Multiply AND assignment operator. It multiplies right operand with the left operand and assign the result to left operand.	c *= a is equivalent to c = c * a
/=	Divide AND assignment operator. It divides left operand with the right operand and assign the result to left operand.	c /= a is equivalent to c = c / a
%=	Modulus AND assignment operator. It takes modulus using two operands and assign the result to left operand.	c %= a is equivalent to c = c % a
**=	Exponent AND assignment operator. It performs exponential (power) calculation on both operands and assign value to the left operand.	c **= a is equivalent to c = c ** a
//=	Floor Division AND assigns a value. It performs floor division on both operands and assign value to the left operand.	c //= a is equivalent to c = c // a

### ► Bitwise operators

- Bitwise operator works on bits and perform bit by bit operation. Assume if a = 65; and b = 12; Now in binary format they will be as follows:

a = 0100 0001  
b = 0000 1100

- The table below lists the bitwise operators in Python. Let's assume a=65 and b=12 for this example.

Operator	Description	Example
&	Binary AND operator copies a bit to the result if it exists in both operands.	(a & b) results in 0 (0000 0000)
	Binary OR operator copies a bit if it exists in either operand.	(a   b) results in 77 which is 1001101
^	Binary XOR operator copies the bit if it is set in one operand but not both.	(a ^ b) results in 77 which is 1001101
~	Binary ones complement operator is unary and has the effect of 'flipping' bits.	(~a) results in -66 which is 10111110.
<<	Binary Left Shift operator. The left operands value is moved left by the number of bits specified by the right operand.	a << 2 results in 260 which is 100000100
>>	Binary Right Shift operator. The left operands value is moved right by the number of bits specified by the right operand.	a >> 2 results in 16 which is 10000

### ► Logical Operators

- There are following logical operators supported by Python language.

Operator	Description	Example
AND	Logical AND operator. If both the operands are true then condition becomes true.	a=5 b=10 (a and b) is true.
OR	Logical OR operator. If any of the two operands are non zero then condition becomes true.	(a or b) is true.
NOT	Logical NOT operator. Use to the logical state of its operand. If a condition is true then Logical NOT operator will make false.	not(a) is false.

### ► Membership Operators

Python has membership operators, which test for membership in a sequence, such as strings, lists, or tuples. There are two membership operators explained below:

Operator	Description	Example
in	Evaluates to true if it finds a variable in the specified sequence or false otherwise.	y = [] if x in y: returns 1 if x is a member of sequence y.
not in	Evaluates to true if it does not find a variable in the specified sequence or false otherwise.	y = [] if x not in y: here not in results in a 1 if x is not a member of sequence y.

### ► Identity Operators

Identity operators compare the memory locations of two objects. There are two identity operators explained below:

Operator	Description	Example
is	Evaluates to true if the variables on either side of the operator point to the same object or false otherwise.	x is y, here <b>is</b> results in 1 if id(x) equals id(y).
is not	Evaluates to false if the variables on either side of the operator point to the same object or true otherwise.	x is not y, here <b>is not</b> results in 1 if id(x) is not equal to id(y).

### ► Operator precedence

The following table lists all operators from highest precedence to lowest.

Operator	Description
**	Exponentiation (raise to the power)
~ + -	Complement, unary plus and minus (method names for the last two are +a and -a)
* / % //	Multiply, divide, modulo and floor division
+ -	Addition and subtraction
>> <<	Right and left bitwise shift
&	Bitwise 'AND'
^	Bitwise exclusive 'OR' and regular 'OR'
<= < > >=	Comparison operators
<> == !=	Equality operators
= %= /= //= -= += *= **=	Assignment operators
is, is not	Identity operators
in not in	Membership operators
not or and	Logical operators

## Topic-2 Errors



### Revision Notes

- A bug is an error in the code.
- Three types of errors can occur compile time errors, run-time errors and logical errors.
- Compile time error occurs at the time of program compilation.
- Compile time errors can syntax errors or semantic errors.
- Run time errors occur during the execution of a program and are also known as exceptions.
- Logical errors occur when a desired result is not obtained from a program. This is due to some mistake in the program logic.
- Debugging is the process of finding errors in a program.

## CHAPTER-8

## CONDITIONAL AND LOOPING CONSTRUCTS



### Revision Notes

- **Flow of Control:** It is the order of execution of various statements written in a program.
- The control and flow of the program can be divided into:

- **Sequence** : When the statements are executed in a sequence one after the other.
- **Selection/Conditional/Decision** : When the statements are executed depending on the validation of the condition.
- **Iteration/Looping** : When a given set of statements are executed more than once due to presence of a loop.

► **Types of Selection Statements:**

- **if Statement :**
  - ♦ Simplest form of decision statement.
  - ♦ Syntax
 

```
if condition:
    set of statements
```
  - ♦ The set of statements will be executed only if the condition is true.
  - ♦ The set of statements can have any number of statements.
- **if-else Statement :**
  - ♦ Allows two different sets of statements to be executed upon satisfaction or non-satisfaction of the test condition.
  - ♦ Syntax
 

```
if condition:
    set1 of statements
else:
    set2 of statements
```
  - ♦ The set1 of statements will be executed if the condition tested is true.
  - ♦ The set2 of statements will be executed if the condition tested is false.
  - ♦ The two sets of statements can have any number of statements.
- **if-elif-else Statement**
  - ♦ Allows simultaneous testing of more than one conditions and also allows different sets of statements to be executed upon satisfaction or non-satisfaction of the test condition.
  - ♦ **Syntax**

```
if condition1:
    set1 of statements
elif condition2:
    set2 of statements
elif condition3:
    set3 of statements
elif condition4:
    set4 of statements
else:
    setN of statements
```
  - ♦ The set1 of statements will be executed if test condition1 is true. In case, condition 1 is false, then program flow will move to elif and tests condition 2.
  - ♦ The set2 of statements will be executed if test condition2 is true. In case, condition 2 is false, then program flow will move to next elif and tests condition 3.
  - ♦ The set3 of statements will be executed if test condition3 is true. In case, condition 3 is false, then program flow will move to next elif and tests condition 4.
  - ♦ The set4 of statements will be executed if test condition4 is true. In case, condition 4 is false, then program flow will move to next elif and tests the next condition.
  - ♦ When all the conditions are false, the program flow will move to the else and its corresponding set of statements will be executed.
- **Nested if-else Statement:**
  - ♦ Allows a conditional statement inside another conditional statement such as *if* or *else* or *elif* statement inside another *if* or *elif* or *else* statement.
  - ♦ Syntax

```

if condition1:
    set1 of statements
if condition11:
    set11 of statements
elif condition12:
    set12 of statements
else:
    set13 of statements
elif condition2:
    set2 of statements
else:
    set3 of statements

```

#### ► Types of Looping Statements:

- **for Loop / definite Loop:**

- ♦ Used to repeat the set of instructions over a defined range of values.
- ♦ The condition for execution of loop is checked after every repetition.
- ♦ else statement is executed when the control exits the for loop.
- ♦ Syntax  

```

for <Control_variable> in <range of values>:
    set1 of statements
else:
    set2 of statements

```
- ♦ range( ) function - used to create a list of values to be taken by loop with a start, stop and increment (optional) value. Syntax: range (start, stop[, increment] )

- **while Loop / Indefinite Loop :**

- ♦ Use to repeat the set of instructions till the test condition is true.
- ♦ The condition for execution of loop is checked after every repetition.
- ♦ else statement is executed when the condition becomes false.
- ♦ Syntax:  

```

while <condition>:
    set1 of statements
else:
    set2 of statements

```

- **Infinite loop:** It is a loop in which the test condition can never become false.

- **Nested loop:** When one loop is placed inside another loop. First, the condition of outer loop is tested, if it is true then condition of inner loop will be tested.

- **Syntax of nested while loop**

```

while condition:
    while condition:
        set1 of statements
        set2 of statements
        set3 of statements

```

- **Syntax of nested for loop**

```

for <Control_variable1> in < range of values>:
    for <Control_variable2> in < range of values>:
        set1 of statements
        set2 of statements
        set3 of statements

```

➤ **Jump statements:** These statements are used to take the control of the program out of the loop even if the test condition is still true.

➤ **Types of Jump Statements:**

- **Break statement** – It is used to stop execution of the loop immediately and transfer the flow of control to the statement immediately after the loop.

Syntax

break

- **Continue statement** – It is used to stop execution of the loop immediately and transfer the flow of control to the beginning of the loop again.

Syntax

continue

## CHAPTER-9

# STRINGS



### Revision Notes

➤ **String** is a consecutive sequence of characters which are enclosed in single quotes (') or double quotes (").

➤ An **Empty String** is a string that has no characters.

➤ Quotes can be inserted inside a string by placing \ before opening and closing quotes.

➤ \n is used to display the output on the next line of the screen.

➤ \t is used to tabulate the output.

➤ Multiline strings are represented by enclosing the text in triple quotes (""").

➤ **Traversing** a string means accessing all the elements of a string.

➤ Each character in a string has an index value.

➤ Individual elements of a string can be accessed by enclosing index value in [ ].

➤ Using float or other types for index results in TypeError.

➤ Left index of a string starts from 0 and right index starts from -1.

➤ Each character of a string can be accessed using a for loop.

**Syntax:** for i in string\_name:

print(i)

➤ String operators

- **Concatenation** – Creating a new string by adding two strings using +.

**Syntax:** string3 = 'string1' + 'string2'

- **Replication** – Repeating a string from multiple copies of same string using \*.

**Syntax:** string2 = integer \* 'string1'

- **Membership** – Checks whether a particular character exists in a string or not using 'in' or 'not in'.

**Syntax:** <substring> in/not in <string1>

- **Comparison** – Compares two strings on the basis of ASCII value of the characters using <, >, <=, >=, ==, !=.

**Syntax:** string1 <= string2

➤ **Slicing** – Retrieves a subset of the string.

**Syntax:** string\_name [start:end]

➤ Slicing means taking out a substring from a string.

- Extending indexing can be used in place of the slice object.

**Syntax:** string [start:end:step]



- Content of the strings cannot be changed after creation. Modifying contents of string by replacing / deleting/ adding a character shows TypeError.

- **Functions in strings**

- **len()** – Returns the length of the string.  
**Syntax:** len(string\_name)
- **capitalize()** – Returns copy of the string with first letter in uppercase.  
**Syntax:** string\_name.capitalize()
- **split()** – Breaks a string into substrings at the specified place.  
**Syntax:** string\_name.split([separator [, maxsplit]])
- **replace()** – Replaces all occurrences of the old string with the new string.  
**Syntax:** string\_name.replace(old string, new string)
- **isalnum()** – returns true if all the characters of the string are alphanumeric.  
**Syntax:** string.isalnum()
- **isalpha()** – returns true if string contains only alphabet.  
**Syntax:** string.isalpha()
- **isdigit()** – returns true if string contains only digits and false otherwise.  
**Syntax:** string.isdigit()
- **islower()** – returns true if all the letters that the string contains are in lower case.  
**Syntax:** string.islower()
- **isupper()** – returns true if all the letters that the string contains are in upper case.  
**Syntax:** string.isupper()
- **isspace()** – returns true if string contains only whitespace characters.  
**Syntax:** string.isspace()
- **lower()** – converts all uppercase characters in a string to lower case.  
**Syntax:** string.lower()
- **upper()** – converts all lowercase characters in a string to uppercase.  
**Syntax:** string.upper()
- **title()** – returns a copy of the string in title case.  
**Syntax:** string.title()
- **lstrip()** – returns a string with all the leading whitespaces(default) removed. Can also be used to remove any other leading character.  
**Syntax:** string.lstrip([chars])
- **rstrip()** – returns a string with all the trailing whitespaces(default) removed. Can also be used to remove any other trailing character.  
**Syntax:** string.rstrip([chars])
- **strip()** – returns a string with all the leading and trailing whitespaces(default) removed. Can also be used to remove any other leading and trailing character.  
**Syntax:** string.strip([chars])
- **count()** – returns the number of occurrences of a substring in the given string, counting from the start position till the end position if specified.  
**Syntax:** string.count(substr, start =0 , end=len(string))
- **find()** – returns the index if a string occurs in the given string, counting from the start position till the end position if specified. If no occurrence is found it returns -1.  
**Syntax:** string.find(substr, start=0, end = len(string))
- **index()** – is same as find() except that it raises an error if the string is not found.  
**Syntax:** string.index(substr, start=0, end =len(string))
- **partition()** – searches for a string in the given string and returns a tuple containing three elements :
  1. The part of the string before the string to be searched.
  2. The search string.

3. Part of the string after the search string.

**Syntax:** string.partition(value)

- **endswith()** – determines if a string or a substring ends with suffix; returns true, if so and false otherwise.

**Syntax:** string.endswith(suffix, start, end)

- **startswith()** – determines if a string or a substring of string starts with substring; returns true, if so and false otherwise.

**Syntax:** string.startswith (str, beg=0, end=len(string))

- **join()** – returns a string in which the element of sequence have been joined by string separator.

**Syntax:** string.join(iterable)

## CHAPTER-10

### LISTS



#### Revision Notes

#### LISTS

- **List** is a collection of values like series or an ordered sequence of values.
- The items in a list can be of any type such as string, integer, float, object, etc.
- Elements of a list are depicted in square brackets [ ], separated by commas.
- Values in the list can be modified in place because it is mutable.
- The values that make up a list are called its **elements**.
- **Syntax** for creating a list:  
`<list_name> = [ ]`
- A list with blank or no values is called an **empty list**.
- **Creating a list from an existing sequence:**
  - Creating a list from a sequence  
`<new_list_name> = list (sequence)`
  - Creating an empty list  
`list_name = list( )`
- List can also be created through user input.
- List index can be a positive or negative integer value.
- An **IndexError** appears if the user tries and accesses elements that do not exist in the list.
- In **Python**, **lists** are mutable. It means that the contents of the list can be changed after it has been created.
- **Traversing a list** means accessing each element of a list.
- List Traversal Using for Loop :*
  - Using 'in' operator  
`for i in <list_name>`  
`print (i)`
  - Using range( ) function  
`for i in range(len<list_name>):`  
`print (list_name[i])`
- *List Traversal Using while Loop:*  
`while i < len(list_name):`  
`print(list_name[i])`  
`i += 1`
- If we assign the elements of one list to another list, both shall refer to the same object.
- Changes made with one alias get reflected in the other alias.
- In **Python**, while comparing two lists, each element is individually compared in lexicographical order.

► Two lists can be compared if they are of comparable type, otherwise Python flashes an error.

► **Operations on Lists:**

- **Concatenation** – A process in which multiple lists can be combined together with the help of certain operators using '+'.  
**Syntax:**  
`list3 = list1 + list2`

- **Replication** – A process in which a list gets replicated or repeated a specific number of times using '\*'.  
**Syntax:**  
`list1 * 3`

- We cannot multiply two lists.

- **Membership Operators** – An operation carried out to check whether a particular element is a member of that list or not.

- Using 'in' operator – returns "True" if the element appears in the list, otherwise returns "False".

- Using 'not in' operator – returns "True" if the element does not appear in the list, otherwise returns "False".

**Syntax:**

`print (<element> in/not in <list1>)`

- **Indexing** – An index value is assigned for each item present in the sequence.

- In Python, indexing starts from 0.

- Negative indices identify positions from the end of the list.

- **Slicing** – It is an operation in which the user can slice a particular range from that sequence.

- List slices are sub-part of a list extracted out.

**Syntax:**

`list1 [start: stop: step]`

► **Built-in Functions for lists:**

- **append()** – Adds a single item to the end of the list and does not create a new list.

**Syntax:**

`list1.append(item)`

- **extend()** – Adds one list at the end of the other list.

**Syntax:**

`list1.extend(list2)`

- **insert()** – Inserts an element at a specified index.

**Syntax:**

`list1.insert(index_number, value)`

- **reverse()** – Reverses the order of the elements in a list.

**Syntax:**

`list.reverse()`

- **index()** – Returns the index of first matched item from the list.

**Syntax:**

`list.index(<item>)`

- **Add/Change List Elements** – Changes an item or a range of items using '='.

**Syntax:**

`list[index] = <new value>`

- **len()** – Returns the length of the list.

**Syntax:**

`len(list)`

- **sort()** – Sorts the items of the list in place.

- For ascending order

**Syntax:**

`list.sort()`

- For descending order

**Syntax:**

`list.sort(reverse = True)`

- **clear()** – Removes all items from the list.

**Syntax:**

```
list.clear()
```

- **count( )** – Counts how many times an element has occurred in a list and returns it.

**Syntax:**

```
list.count(element)
```

- **sorted( )** – It takes a list as parameter and creates a new list consisting of the same elements arranged in sorted order.

```
>>> list1 = [23,45,11,67,85,56]
```

```
>>> list2 = sorted(list1)
```

```
>>> list1
```

```
[23, 45, 11, 67, 85, 56]
```

```
>>> list2
```

```
[11, 23, 45, 56, 67, 85]
```

➤ **Deletion operation** – For deleting an item from a list.

- If index is known

- **pop( )** – Removes the element from specified index and returns the removed element.

**Syntax:**

```
list.pop(index)
```

- If no index value is provided, the last element in the list is removed.

- **del statement** – Removes the specified element but does not return the removed element.

**Syntax:**

```
del list(index)
```

- to delete single element

OR

```
del list(start index : stop index)
```

- to delete a range of elements

- If element is known but its index is not known

- **remove( )**

**Syntax:**

```
list.remove(element)
```

➤ **Searching the list**

- For a particular element / index – **index( )**

**Syntax:**

```
list.index(element)
```

- For the maximum value in the list – **max( )**

**Syntax:**

```
max(list)
```

- For the minimum value in the list – **min( )**

**Syntax:**

```
min(list)
```

➤ A list that contains another list as element is known as **nested list**.

```
list1 = [1,2,'a','c',[6,7,8],4,9]
```

#fifth element of list is also a list

```
>>> list1[4]
```

```
[6, 7, 8]
```

➤ Individual items in a nested list can be accessed using multiple indexes.

➤ **list()** is used to create a list or to make copy of a list.

➤ **sum()** adds up all the elements of a list.

➤ **mean()** is used to calculate mean of a given list of numbers.

➤ Linear search on a list can be performed by checking each element starting from index ( ) and continuing the search until the number is found.

➤ Frequency of elements of a list can be found using `collections.counter()`; it returns a dictionary with elements of list as key and frequency as values element.

# CHAPTER-11

## TUPLES



### Revision Notes

- A **tuple** consists of group of values separated by commas.
- Element of Tuples are depicted enclosed within parentheses ( ).
- The values that make up a tuple are called its **elements**.
- Elements in a tuple need not be of the same type.
- The index value of tuple starts with 0.
- Tuples are faster and more efficient than lists.
- If a tuple comprises of a single element, the element should be followed by a comma. Such a tuple is called a **singleton** tuple.

**Creating tuple with a single element:**

**Syntax:**

```
tuple_name = ("January",)
```

**OR** tuple\_name = tuple( )

- A tuple can be created by accepting input by user input using *while* loop.
- Elements of a tuple can be accessed in the same way as a list or string using indexing and slicing. Each element in the tuple is accessed by the index in the square brackets []. An index starts with zero and ends with (number of elements - 1), as shown below.

```
nums = (1, 2, 3, 4, 5)
print(nums[0]) # prints 1
print(nums[1]) # prints 2
print(nums[4]) # prints 5
```

The tuple supports negative indexing also, the same as list type. The negative index for the first element starts from -number of elements and ends with -1 for the last element, as shown below.

```
names = ('Jeff', 'Bill', 'Steve', 'Yash')
print(names[-4]) # prints 'Jeff'
print(names[-3]) # prints 'Bill'
print(names[-2]) # prints 'Steve'
print(names[-1]) # prints 'Yash'
```

- Tuples can be nested. This means that tuples can be placed inside other tuples.
- The individual elements of a tuple can be accessed through their indices given in square brackets [ ].
- A tuple can be traversed using:
  - **'in' operator with for loop.**

**Syntax:**

```
for i in tuple_name:
    print(i)
```

- **range( ) function**

**Syntax:**

```
for i in range(len (tuple_name)):
    print(tuple_name [i])
```

**Slicing is used to retrieve a subset of values.**

**Syntax:**

```
tuple_name[start: stop: step]
```

- Two tuples can be combined together using '+' operator.



- The elements of a tuple can be repeated using '\*' operator.
- The users can check whether a particular element is a member of that tuple or not.
  - Using 'in' operator – returns "True" if the element appears in the tuple, otherwise returns "False".
  - Using 'not in' operator – returns "True" if the element does not appear in the tuple, otherwise returns "False".

**Syntax:**

<element> in/not in <tuple\_name>

- **Slice Operator:** Tuple\_name[start:end] will give you elements between indices start to end-1.
 

```
>>> alpha=('q','w','e','r','t','y')
>>> alpha[1:-3]
('w', 'e')
```

➤ **Tuple functions:**

- **len()** – Returns the length of a tuple.

**Syntax:**

len(tuple\_name)

- **count()** – Counts the occurrence of an item in the tuple.

**Syntax:**

tuple\_name.count(element)

- **any()** – Returns True if a tuple is having at least one item and returns False if the tuple is empty.

**Syntax:**

any(tuple\_name)

- **max()** – Returns the element with maximum ASCII value in the tuple.

**Syntax:**

max(tuple\_name)

- **min()** – Returns the element with minimum ASCII value in the tuple.

**Syntax:**

min(tuple\_name)

- **sorted()** – Sorts the elements of a tuple.

**Syntax:**

sorted(tuple\_name)

- **index()** – Finds the first index of a given item and returns the index.

**Syntax:**

tuple\_name.index(value, start, end)

- Tuples can be compared using comparison operators like <, >, =, !=, etc.
- In Python, comparison operators start by comparing the first element from each sequence. If they are equal, it goes on to the next element until it finds the elements that differ. The subsequent elements are not considered.
- A tuple is deleted using del statement.

**Syntax:**

del tuple\_name

- A tuple can contain other compound objects such as lists, dictionaries and even another tuple. A tuple that contains another tuple as an element is known as a **nested tuple**.
- Each of the member tuple is treated as a single element for the main tuple, e.g., the tuple (1, 2, ('A', 'B', 'C'), 3, ('D', 'E')) has 5 elements.
 

```
>>> T=(5,10,(4,8))
>>> T
(5, 10, (4, 8))
```
- tuple() is used to create a tuple.
- sum() finds sum of all the elements of a tuple containing numbers.



## Key Terms

- **Packing:** Forming a tuple from individual elements is called packing.
- **Unpacking:** Creating individual values from elements of a tuple is called unpacking.
- Assignment of tuple is a useful feature in Python. It allows a tuple of variables on the left side of the assignment operator to be assigned respective values from a tuple on the right side. The number of variables on the left should be same as the number of elements in the tuple, as shown below.

```
>>> (num1,num2) = (10,20)
>>> print(num1)
10
>>> print(num2)
20
```

## CHAPTER-12 DICTIONARY



## Revision Notes

- **Python Dictionary** is an unordered collection of items where each item is a key-value pair.
- A dictionary can be created by placing items inside curly braces { } separated by a comma.
  - Syntax:**

```
<dictionary_name> = {'key1': 'value1', 'key2': 'value2', ..., 'keyn': 'valuen'}
```
- To access elements in a dictionary, square brackets [ ] along with the key are used.
- Dictionaries are mutable which implies that the contents of the dictionary can be changed after it has been created.
- To add new elements to an existing dictionary
  - Syntax:**

```
dictionary_name ['key'] = 'value'
```
- To modify existing key-value pair in a dictionary
  - Syntax:**

```
dictionary_name ['key'] = 'value'
```
- Traversing a dictionary means accessing each element of a dictionary.
  - Syntax:**

```
for i in dictionary_name:
    print(i, ': ', dictionary_name[i])
```
- To merge two dictionaries
  - Syntax:**

```
dictionary_name.update(dictionary2)
```
  - When two dictionaries are merged, the values of the same key are overwritten.
- To remove an item from the dictionary
  - Using del command
    - Syntax:**

```
del dictionary_name[key]
```
  - Using pop() method
    - Syntax:**

```
dictionary_name.pop(key)
```
- The users can check whether a particular key is present in a dictionary or not.
  - Using 'in' operator – returns "True" if the key is present in the dictionary, otherwise returns "False".
  - Using 'not in' operator – returns "True" if the key is not present in the dictionary, otherwise returns "False".
  - Syntax:**

```
<key> in/not in <dictionary_name>
```
- **Dictionary functions:**
  - **len()** – Returns the number of key-value pairs in the dictionary.

**Syntax:**`len(dictionary_name)`

- **clear()** – Removes all items from the dictionary.

**Syntax:**`dictionary_name.clear()`

- **get()** – Returns the value of a given key in the dictionary.

**Syntax:**`dictionary_name.get(key)`

- **items()** – Returns all the key-value pairs in the dictionary.

**Syntax:**`dictionary_name.items()`

- **keys()** – Returns the list of keys used in the dictionary.

**Syntax:**`dictionary_name.keys()`

- **values()** – Returns the list of values defined in the dictionary.

**Syntax:**`dictionary_name.values()`

- **dict()** function creates a dictionary.
- **del()** removes keys from a dictionary.
- **fromkeys()** creates a new dictionary from the given sequence of elements.

**Syntax:**`dict.fromkeys(sequence[,value])`

It returns a new dictionary.

- If value specified in the **fromkey()** is a mutable object(list etc) then on its updating each element of the sequence is updated.
- **copy()** method is used to create a shallow copy of a dictionary.

**Syntax:**`dict1=dict2.copy()`

- If values associated with dictionary keys are iterable objects, then if the dictionary is modified the other one will too be modified.
- **popitem()** removes the last inserted item from dictionary and returns it as a tuple. It raises a **KeyError** if dictionary is empty.
- **setdefault()** returns the value of a key. If the key is not present it is inserted to the dictionary.
- **max()** with **dict.get** is used to find the key with maximum value in a dictionary.

**Syntax:**`max(iterable, key =dict.get)`

- **max()** and **dict.values()** is used to find the maximum value in a dictionary.

**Syntax:**`max(dict.values())`

- **min()** together with **dict.values()** is used to find the minimum value in a dictionary.
- **sorted()** returns the sorted list of keys in a dictionary.

## CHAPTER-13

# INTRODUCTION TO PYTHON MODULES



### Revision Notes

- The act of partitioning a program into individual components is called 'Modularity'.
- A module is a separately saved unit whose functionality can be reused at will. A Python module has the **.py** extension.
- A Python module can contain objects like docstrings, variable, constants, classes, objects, statements, functions.
- The Python modules that come preloaded with Python are called "standard library modules".

- Python module can be imported in a program using import statement.
- **Importing module:** There are two forms of import statements:
  - (i) `import <module name>`
  - (ii) `from <module> import <object>`
- The math module of Python provides math functionality.
- **ceil (x)** returns the ceiling of x as a float, the smallest integer value greater than or equal to x.
- **floor (x)** returns floor of x as a float, the largest integer value less than or equal to x.
- **fabs (x)** returns the absolute value of x.
- **pow (x, y)** returns x raised to the power y.
- **sqrt (x)** returns the square root of x.
- **cos (x)** returns the cosine of x radians.
- **sin (x)** returns the sine of x radians.
- **tan (x)** returns the tangent of x radians.
- **math.pi** gives mathematical constant  $p = 3.141592...$
- **math.e** gives the mathematical constant  $e = 2.718281...$
- Random number can be generated using random module.
- Random number generators can be used by including random module using the statement `import random`.
- **random()** generates a floating-point number between the range [0.0, 1.0]
- **randint(a, b)** generates a random integer in the range [a, b]. Here the range includes both a & b.
- **randrange()** returns a randomly selected element from the specified range.

**Syntax:**

`random.randrange (start, stop, step)`

- **Statistics module** provides functions to mathematics statistics of numerical data.
- It is imported using the statement `import statistics`.
- **mean()** is used to calculate arithmetic mean of the numbers in the list.
- **median()** is used to return middle value of the numeric data set.
- **mode()** returns the most common data that occurs in the list.

## UNIT-III : SOCIETY, LAW AND ETHICS

### CHAPTER-14

## DIGITAL FOOTPRINTS & SOCIETY AND NETIZEN



### Revision Notes

- **Digital Footprint**
  - Whatever a person be on internet creates his usage or we can say left a shadow behind of that activity and all these activities shadow creates your identity, this identity is called digital footprint. Digital footprint is nothing but the record of what a person do online.
  - Digital footprint includes e-mail you sent, information you shared, websites you visited and the activities you took part online.
  - Digital footprint is used for several reason for example marketers use your digital footprint to find out what kind of product user have interested and a interviewer what kind of activities the candidates perform online, it gives better idea about the candidate's personality.
- Digital footprint is of two types:**
  1. **Active Digital Footprint:** When a user knowingly share the personal data in order to share information about the user by means of social networking digital platform. e.g., when user makes a comment or post something on social media.
  2. **Passive Digital Footprint:** When the personal data of the user are collected without letting him know or collection of personal data of user without the permission of him is known as passive digital footprint. e.g., when user visits any website traces his physical location using user's device IP address.
- **How can you make your digital footprint positive?**

You can make your digital footprint positive by being little careful when you are surfing. You can do following things:

1. Always check the content you post on web. If you are not careful about what you do online then there is the possibility of your negative digital footprint.
2. Don't keep the attitude that you can do anything online and no one will come to know. This information can be fetched even without your permission. All the information stored in the form of cookies which can be accessed by the one who wants to get information about your digital footprint.
3. You should always know what you exactly want because the confusion can create wrong footprint and can mislead the one who is looking for your digital footprint.

➤ **Advantages of Digital Footprint:** Although when we hear the name digital footprint a negative image comes in our mind because it keeps the record of each and every activity of what we do online. But digital footprint has so many advantages:

- Digital footprint makes our online experience friendly. Digital footprint decreases the complexity of the online world. Without digital footprint, our net surfing may not be easy as it is today.
- Digital footprint helps Google Ad Sense to serve as best. Using our digital footprint, Google Ad Sense provides us appropriate advertisement. If they haven't used our digital footprint this may happen that we are searching for job information and they feeding our search with railway booking information.
- These days most of the devices are GPS enabled which add our location to our digital footprint, online maps provides you best route possible.

➤ **Digital Society and Netizen**

- These days, most of the tasks we do are done using digital technologies. We chat online, shop online, learn online, and so much more. As a result, we have quite the online presence. Therefore, in order to maintain a certain code of conduct as one would do so in regular life, there are certain net etiquettes or netiquettes that have been created to ensure that everyone is a responsible digital citizen (netizen).

**Netiquette** is a way to communicate over Internet. In real world, We use a manner to talk so that the exact meaning could successfully convey to the listener. On Internet, this manner is known as Netiquettes which help the user to get exact idea of what is said.

- Netiquettes makes the communication process successful because without it sender's message can be misinterpreted by the receiver though he will not be able to see the facial expression of the sender.
- Netiquettes makes the communication more effective by adding human emotion to it. By using netiquettes, sender can express what he was feeling while writing the post or message.
- Netiquettes help the user in establishing good relation with the other users because when use netiquettes he can express what he feels or what he wants to say.
- Netiquettes ensure careful word formation and also placing them at right place. These also ensure the correct use of emotions.
- Netiquettes help in making good digital footprint of yours. If you are applying netiquettes while talking to friends or other person then you are making good reputation of yourself.

➤ **One should be ethical, respectful and responsible while surfing the Internet.**

- **Ethics**

1. **Copyright Violation:** Do not use copyrighted information without proper license i.e. permission of the creator/owner. Creators such as Q-Programming have put a lot of effort into sharing their content and hence one should respect these efforts and not use materials without proper authorization.
2. **Sharing Information:** With access to unlimited online resources, we should share useful information with others. However, make sure the information you share is accurate and from a reputable source. A lot of fake information is spread nowadays which can lead to consequences.

- **Respect**

1. **Privacy:** Respect the privacy of your fellow netizens. Good digital citizens understand that everyone has the same rights and freedoms. If someone shares private information with you DO NOT send it to anyone else without the sender's consent.
2. **Diversity:** Public forums and chat sites have great diversity in contributing members. Make sure you respect other people's cultures and beliefs and expect that they do the same for you.

- **Responsibility**

1. **Cyber Bullying:** Remember that anything you say online to someone has the same effect as if you say it to their face. Many people say offensive and hurtful things online thinking that they can't be held accountable. However, remember that everything you say is part of your digital footprint and can be easily



traced back to you. Therefore, be careful what you say on public forums as you can get in trouble, but also, most importantly, you can cause harm to others.

2. **Trolling:** Internet trolls are people who intentionally disrupt harmony on online platforms. They thrive on attention and hence you should not fuel them by supporting them or rebutting. Trolls can be like online bullies they have fun when they make someone react to their comments. The best thing you can do is ignore them.

**Following are the Do's and Don'ts that one should follow while working on the Internet.**

Do's	Don'ts
Keep message and posts brief..	Post inflammatory/offensive comments.
Reread your posts or emails to make sure they say what you intend.	Write in ALL CAPS. It is considered as SHOUTING on the net.
Remember that you leave a digital footprint. So, be careful what you write.	Respond to internet trolls/personal attacks.
Use discretion.	Post private or embarrassing images/comments.
Include subject line in an email.	Use sarcasm. It might be misinterpreted.
Protect personal information.	Violate copyright laws. Make sure your work is your own or properly cited.
Obey copyright laws.	Exclude people or talk behind their backs.
Stay focused and stick to the topic.	Spam others by sending large number of unsolicited emails.

#### ➤ **Communication Etiquettes**

- As a human being, communication is the basic form of talking to each other and passing on a message.
- Communication etiquettes refers to the rules or etiquettes followed while sending emails, chatting, sending SMS, call, posting on forums and social media.

#### ➤ Good communication over email, chat room and other such forums require a digital citizen to abide by the communication etiquettes such as:

- **Precision**
  1. **Time:** Everyone is busy and therefore we should not waste our valuable time on unnecessary emails and comments. Similarly, do not expect others to respond to you instantly as they may also be preoccupied.
  2. **Data Limits:** Avoid sending large files over the internet as they use a lot of data and bandwidth, which come with their respective costs. You should make it a habit to send compressed files or links, using various free services.
- **Politeness:** While talking to someone online in a synchronous/real-time or asynchronous format, be polite and non-aggressive. Do not use slurs or abuses even if you get worked up.
- **Credibility:** As discussed in a previous point, make sure whatever you write on the internet is accurate and fact-checked. Spreading false information leads to unnecessary confusion.

#### ➤ **Social Media Etiquettes**

- In the current digital era, we are familiar with different kinds of social media and we may have an account on Facebook, Google+, Twitter, Instagram, Pinterest, or YouTube channel.
- Social media are websites or applications that enable their users to participate in social networking by creating and sharing content with others in the online community.

#### ➤ In social media too, there are certain etiquettes we need to follow such as:

- **Security**
  1. **Passwords:** Choose strong passwords to safeguard against data leaks or hackers. Use a combination of numbers, alphabets, allowed symbols and cases to create a password which is tough to crack. Also, avoid sharing your account information with others.
  2. **Friends:** Social networking sites allow us to talk to anyone else on the platform. Be wary, however, as not everyone is friendly. Some people may even have malicious intentions. Be careful of who you interact with online and only converse with those who you have met in real life
  3. **Fake Information:** There is a lot of fake information on the internet. This information is easily spread via social networking sites. Therefore, when you see something, make sure it is accurate. Experienced users are able to distinguish fake news from real news. Those who don't can verify information from reputed sources. Although platforms have algorithms to fact check, report any misleading information to avoid further spread.
- **Reliability:** When you are about to upload something to the internet, pause for a moment and think of the consequences your actions could have in the future. Once you are sure what you are posting is acceptable, do so. Remember, as discussed earlier, that even if you delete an upload, it will still be stored on the service's servers.

# CHAPTER-15

## DATA PROTECTION AND IT ACT



### Revision Notes

#### ► Data Protection

- It is the process of safeguarding important information from corruption, compromise or loss. In short, you should be able to decide whether or not you want to share some information, who has access to it, for how long, for what reason and be able to modify some of this information and more.
- A large part of a data protection strategy is ensuring that data can be restored quickly after any corruption or loss. Protecting data from compromise and ensuring data privacy are other key components of data protection. There are two key areas of data management used in data protection as:
  - (i) Data life cycle management is the process of automating the movement of critical data to online and offline storage.
  - (ii) Information life cycle management is a comprehensive strategy for valuing, cataloguing and protecting information assets from application and user errors, malware and virus attacks, machine failure or facility outage and disruptions.

#### ► Purpose of data protection

- Storage technologies that can be used to protect data include a disk or tape backup that copies designated information to a disk based storage array or a tape cartridge device so it can be safely stored.
- Mirroring can be used to create an exact replica of a website or files so they are available from more than one place.
- Storage snapshots can automatically generate a set of pictures to information stored on tape or disk, enabling faster data recovery while continuous data protection backs up all the data in an enterprise whenever a change is made.

#### ► Ways to Protect Data

- Know exactly what you have and where you keep it.
- Maintain a list of employees with access to sensitive data.
- Install trustworthy virus/malware protection software and run regular scans.
- Run regular backups of your important and sensitive data.

#### ► Intellectual Property

- When someone owns a house or a motorcycle, we say that the person owns that property. Similarly, if someone comes out with a new idea, this original idea is that person's intellectual property.
- Intellectual property refers to the inventions, literary and artistic expressions, designs and symbols, names and logos. The ownership of such concepts lies with the creator or the holder of the intellectual property.
- This enables the creator or copyright owner to earn recognition or financial benefit by using their creations or inventions.

#### ► Intellectual Property Rights (IPR)

- Intellectual Property Rights are the exclusive rights given to the person over his/her creation for specific time period. These rights allow the patents or owner to buy, sell, and exchange their licensed goods to different people or organisations. Intellectual property rights are largely covered by laws governing to patents.
- Copyrights, industrial design rights, trademarks, plant variety rights, trade dress, geographical indications, circuit design rights and supplementary protection certificates for pharmaceutical products and database rights, etc.
- Intellectual property is legally protected through copyrights, patents, trademarks etc.
  - (i) **Copyright:** It grants legal rights to creators for their original works like writing, photograph, audio recordings, video, sculptures, architectural works, computer software, and other creative works like literary and artistic work.

Copyrights are automatically granted to creators and authors. Copyright law gives the copyright holder a set of rights, that they alone can avail legally. The rights include right to copy (reproduce) a work, right to create derivative works based upon it, right to distribute copies of the work to the public and right to publicly display or perform the work. It prevents others from copying, using or selling the work.

For example, any writer holds the copyright to his book. It would be an infringement of the writer's copyright if someone used parts of his book without permission. To use other's copyrighted material, one needs to obtain a license from them.

- (ii) **Patent:** It is usually granted for inventions. Unlike copyright, the inventor needs to file for patenting the invention. When a patent is granted, the owner gets an exclusive right to prevent others from using, selling or distributing the protected invention.

Patent gives full control to the patentee to decide whether or how the invention can be used by others. Thus it encourages inventors to share their scientific or technological findings with others. A patent protects an invention for 20 years, after which it can be freely used. Recognition and financial benefits right the environment and provide motivation for more creativity and innovation.

- (iii) **Trademark:** It includes any visual symbol, word, name, design, slogan, label etc., that distinguishes the brand or commercial enterprise, from other brands or commercial enterprise.

For example, no company other than ABC can use the ABC brand to sell shoes or clothes. It also prevents others from using a confusingly similar mark, including works or phrases.

#### ► Objectives of IPR

- The policy is a comprehensive document that lays down seven objectives which have been elaborated with actionable steps to be undertaken by the identified nodal ministry or department towards achieving objectives.
  - IPR Awareness:** Outreach and Promotion — To create public awareness about the economic, social and cultural benefits of IPRs among all sections of society.
  - Generation of IPRs:** To stimulate the generation of IPRs.
  - Legal and Legislative Framework:** To have strong and effective IPR laws, which balance the interests of owners rights with larger public interest.
  - Administration and Management:** To modernize and strengthen service oriented IPR administration.
  - Commercialization of IPRs:** Get value for IPRs through commercialization.
  - Enforcement and Adjudication:** To strengthen the enforcement and adjudicatory mechanisms for combating IPR infringements.
  - Human Capital Development:** To strengthen and expand human resources, institutions and capacities for teaching, training, research and skill building in IPRs.
- Intellectual Property laws and enforcement vary widely from jurisdiction to jurisdiction. There are intergovernmental efforts to harmonize them through international treaties such as 1994 World Trade Organisation (WTO) Agreement on Trade Related aspects of Intellectual Property Rights (TRIPs), while other treaties may facilitate registration in more than one jurisdiction at a time.
- With companies, institutions and individuals constantly forging ahead in newer fields and geographical territories and with path breaking inventions becoming the norm, the field of Intellectual Property rights has assumed primordial importance, especially in emerging economics like India.

#### ► Violation of IPR

**Violation of IPR may happen in one of the following ways:**

- Plagiarism:** With the availability of Internet, we can instantly copy or share text, pictures and videos. Presenting someone else's idea or works as one's own idea or work is called plagiarism. If we copy some contents from Internet, but do not mention the source or the original creator, then it is considered as an act of plagiarism. Further, if someone derives an idea or a product, but instead presents it a new idea then also it is plagiarism. It is a serious ethical offense and sometimes considered as an act of fraud. Even if we take contents that are open for public use, we should cite the author or sources to avoid plagiarism.

**Types of Plagiarism:**

- Direct Plagiarism** is the word to word transcription of a section of someone else's work without attribution and without quotation marks.
- Self Plagiarism** occurs when a student submits his or her own previous work or mixes parts of previous work without permission from all professors involved.
- Mosaic Plagiarism** occurs when a student borrows phrases from a source without using quotation marks.
- Accidental Plagiarism** occurs when a person neglects to cite their sources or misquotes their sources.

**Apart from these, other categories of plagiarism are as follows:**

- CLONE:** Cloning is the most problematic and common form of plagiarism.
  - Ctrl + C:** In this type of plagiarism, a particular portion of the text is copied from a single source without any alteration.
  - FIND-REPLACE:** Most common keywords of the copied content are changed.
  - REMIX:** Information from various sources are collected and mixed.
- Copyright Infringement:** Copyright infringement is when we use other person's work without obtaining their permission to use or we have not paid for it, if it is being sold. Suppose we download an image from the Internet and use it in our project. But if the owner of the copyright of the image does not permit its free usage, then using such image even after giving reference of the image in our project is a violation of copyright. Just because it is on the Internet, does not mean that it is free for use. Hence, check the copyright status of writer's work before using it to avoid plagiarism.

- **Trademark Infringement:** Trademark infringement means unauthorised use of other's trademark on products and services. An owner of a trademark may commence legal proceedings against someone who infringes its registered trademark.

#### ► **Software License**

- It is a document that provides legally binding guidelines for the use and distribution of software. Software licenses typically provide end users with the right to one or more copies of the software without violating copyrights.
- **Types of Software License:**
  - (i) Proprietary license is a license where the copyright stays with the producer and the user is granted the right to use the software.
  - (ii) GNU General Public License (GPL) which are agreements under which open source are usually licensed.
  - (iii) End User License Agreement (EULA) indicates the terms under which the end user may use the software.
  - (iv) Creative commons (CC) license is a public copyright license that enables free distribution of a copyrighted work.
  - (v) Apache is the most popular web server software that enables a computer to host one or more websites. It is open source and free to use hence enabling web hosting companies to offer web hosting solutions at minimal costs. Platforms supported by Apache: Linux, Windows and Macintosh operating systems. Apache web server software is also known as Apache HTTP server.

#### ► **Free and Open Source Software (FOSS)**

- FOSS has a large community of users and developer who are contributing continuously towards adding new features or improving the existing features.
- The term free indicates that the software does not have constraints on copyrights. The Term open source refers software development from expert developers collaborating worldwide without any need for reverse engineering. Free/libre open source software (FLOSS) or free open source software (FOSS).
- One major reason for the growth and use of FOSS technology is because users have access to the source so it is much easier to fix faults and improve the applications. In combination with the open license, this simplifies the development process for many enterprises and gives them flexibility that simply is not available with the confiner of a proprietary or commercial product.
- For example, Linux Kernel based operating systems like Ubuntu and Fedora comes under FOSS.
- Some of the popular FOSS tools are office packages, like Libre office, browser like Mozilla Firefox etc.

#### ► **Indian Information Technology Act (IT Act)**

- With the growth of Internet, many cases of cyber crimes, frauds, cyber attacks and cyber bullying are reported.
- The nature of fraudulent activities and crimes keeps changing. To deal with such menaces, many countries have come up with legal measures for protection of sensitive personal data and to safeguard the rights of Internet users.
- The Parliament of India passed its first Cyber law the Information Technology (IT) Act, 2000, on the 17th October 2000, which provides the legal infrastructure for e-commerce in India. The purpose of the
- IT Act, 2000, as mentioned in the language of the Act is: to provide legal recognition for transaction carried out by means of electronic data interchange and other means of electronic communication, commonly referred to as "electronic commerce", which involves the use of alternative to paper based methods of communication and storage of information, to facilitate electronic filling of document with the Government agencies and further to amend the Indian Penal Code, the Indian Evidence Act, 1872, the Banker's Book Evidence Act, 1891 and the Reserve Bank of India Act, 1934 and for matters connected therewith or incidental thereto.
- The General Assembly of the United Nations, by its resolution A/RES/51/162 dated 30th January 1997, adopted the Model Law on Electronic Commerce adopted by the United Nations Commission on International Trade Law. The same resolution recommends among other things that all states give favourable consideration to the Model Law when they enact or revise their law, keeping in mind the need for uniformity of law pertaining to alternatives to paper-based methods of communication and storage of information. The Indian Information Technology Act, 2000, accordingly draws upon the Model Law.
- The implementation of this Act has kick-started a new era of e-governance and will have a lot of impact on the way people do business in India and will also open up new opportunities for E-business as people would be less apprehensive about the legal hassles and issues not under the jurisdiction of law, e.g. authenticity of legal document, hacking, digital signatures and so on.
- Therefore, it is essential for us to understand what the IT Act offers and what are its various perspectives.
- The Government of India's Information Technology Act, 2000 amended in 2008, provides guidelines to the user for the processing, storage and transmission of sensitive information. In many Indian states, there are cyber cells in police stations where one can report any cyber crime. The act provides legal framework for electronic governance by giving recognition to electronic records and digital signatures.



► **Highlights of IT ACT, 2000**

- For a basic understanding of the IT Act by the layman, the salient features of the Act and its relevant portion of an e-business are enumerated below:
- Electronic contracts are legally valid – EDI accorded legal recognition.
- Legal recognition according to digital signature.
- Digital signature to be affected by use of asymmetric crypto system and hash function.
- Security procedure for electronic records and digital signature.
- Appointment of Certifying Authorities (CAs) and the Controller of Certifying Authorities (CC(a) including recognition of foreign Certifying Authorities.
- Controller to be appointed, who will act as repository of all digital signature certificates.
- Certifying Authorities require to get license to issue digital signature certificates.
- Various types of computer crimes defined and stringent penalties provided under the Act.
- Appointment of Adjudicating Officer for holding enquiries under the Act.
- Establishment of Cyber Appellate Tribunal under the Act.
- Appeal from order of Adjudicating Officer to Cyber Appellate Tribunal and not to any civil court.
- Appeal from order of Cyber Appellate Tribunal to High Court.
- Act to apply for offences or contraventions committed outside India.
- Network Service providers not to be liable in certain cases.
- Power to Police officers and other officers to enter into any public place and search and arrest without warrant.
- Constitution of Cyber Regulations Advisory Committee to advise the Central Government and Controller.
- **Amendments effected in:**
  - (a) Indian Penal Code
  - (b) Indian Evidence Act
  - (c) Banker's Books Evidence Act
  - (d) Reserve Bank of India Act

► **Some of the important concepts introduced in the IT ACT, 2000 are:**

- Electronic record
- Secure electronic record
- Digital signature
- Secure digital signature
- Certifying authority
- Digital signature certificate.

## CHAPTER-16

# CYBER CRIME & SAFETY AND SAFELY ACCESSING WEBSITES



### Revision Notes

► **Cybercrime**

- Cybercrime is defined as a crime in which a computer is the object of the crime (hacking, phishing, spamming) or is used as a tool to commit an offence (child pornography, hate crimes).
- Cybercriminals may use computer technology to access personal information, business trade secrets or use the internet for exploitative or malicious purposes. Criminals can also use computers for communication and document or data storage.
- Criminals who perform these illegal activities are often referred to as hackers.
- Common types of cybercrime include online bank information theft, identity theft, online predatory crimes and unauthorized computer access. More serious crimes like cyber terrorism are also of significant concern. Cybercrime encompasses a wide range of activities but these can generally be broken into two categories:
  - (1) Crimes that target computer networks or devices. These types of crimes include viruses and denial-of-service (DoS) attacks.
  - (2) Crimes that use computer networks to advance other criminal activities. These types of crimes includes cyberstalking, phishing and fraud or identity theft.



► The different kinds of cyber crimes are:

- **Hacking:** Unauthorized use of a computer system is known as hacking. Hackers are skilled criminals who know how to exploit hardware and software by identifying weaknesses or bugs. Some hackers, known as ethical hackers, do so with good intentions. They are also known as white hat hackers. They identify weaknesses in computer hardware and software and use their knowledge to uncover any loopholes hackers may find. Hackers with malicious intentions are known as black hat hackers. They hack systems in order to steal valuable information, damage systems, or make money by holding data as a hostage. They use their skills to spread harm through illegal means.
- **Phishing and Fraud Emails:** Phishing is using fake links and emails to extract sensitive data or to install malware on the user's system. This data is misused as per the criminal's desires. Phishing scams can be identified by inspecting links, words and contents to look for errors.
- **Identity Theft:** Besides phishing, criminals steal people's identities. They steal personal information from others illegally and then pose as them in different places. This kind of phishing is primarily done for monetary gains. Criminals can use identity theft for financial gain, avoidance of detection by authorities, and to obtain medical drugs or treatment.
- **Ransomware:** Ransomware blocks users from accessing their own data by encrypting it. It holds data as hostage and releases it only when a ransom has been paid. It can be downloaded when visiting unsecure sites or links.

► Some more common cybercrimes are:

(A) **Cyber Trolls:** These are the persons who start quarrels on the internet to distract a community and provoking the readers to behave emotionally.

(B) **Cyber Bullying:** This is the attack upon an individual through electronic means to abuse or intimidate others. The various forms of cyber bullying are explained as:

- **Doxing:** Publishing, revealing personal information about an individual online, for the purpose of defaming, humiliating or harassing the victim.
- **Harassment:** Threatening by posting, hurtful or intimidating messages online, or sending them directly to someone, with the intention of harming that person.
- **Impersonation:** Creating fake accounts or gaining access to a person's real social media accounts and posting things to damage the victim's reputation.
- **Cyberstalking:** Tracking and monitoring a person's online activity and using the internet to stalk or harass an individual.

(C) **Cyber Stalking** - It is the use of internet to harass an individual or an organisation.

(D) **Spreading Rumours** - This is the act of creating fake e-mail IDs, masking actual identification and posting false information on social media.

► **Combatting and Preventing Cyber Crime**

We can protect ourselves against cyber criminals by:

- Installing antivirus software and conducting regular scans.
- Backing up important data.
- Updating system software regularly as updates fix different bugs.
- Being careful about the sites we visit, links we click, and things we download.
- Using strong passwords (as mentioned earlier), even for the wireless network.
- Not saving personal data on other people's systems.
- Not using cookies for unknown sites.
- Only using secure sites for transactions.

► **Cyber safety**

- Cyber safety refers to the safe and responsible use of information and communication technology.
- Identity theft is a type of fraud that involves using someone else's identity to steal money or gain other benefits.
- Whenever a user visits a website, the browser may reveal the location via the device's IP address.
- Browsers also provide search and browsing history.

► **Websites track their visitors through:**

- **IP Address:** It is a device's unique address to connect to the Internet. From this IP Address, a website can determine the rough geographical location of the device.
- **Cookies:** These are small pieces of information websites can store in the browser.
- HTTP Referer is a link on a web page to outside website. It automatically provides the visitor's information, such as IP address, location, web browser, machine type, etc. to the linked website.
- Super cookies are persistent cookies that come back even after being deleted.
- User Agent is a string or a line of text, that browser sends to every website, a device connects to. It tells the web-server about the web browser and the OS being used.

- Anonymous browsers allow users to view websites without revealing any personal information of the user like their IP address, machine type, location etc.
  - Confidentiality of Information ensures that only authorized users get access to sensitive and protected data.
  - **Confidentiality of information is maintained by:**
    - Encryption of data.
    - Use of firewall
    - Private browsing
    - Accessing safe sites
    - Carefully handling e-mails
    - Avoiding use of public network.
  - **Cyber Forensics:** It is an electronic discovery technique used to determine and reveal technical criminal evidence.
  - The Information Technology Act, 2000 or IT Act, 2000 is the primary law in India dealing with cybercrime and electronic commerce.
  - **Tips for safely browsing the Internet:** Use the following advice when browsing the web to significantly reduce your risk of being a victim of cybercrime:
    - Update the software regularly.
    - Keep the web browser protected.
    - Observe safe online behaviour.
    - Use strong and unique passwords.
    - Download only trusted files and applications.
    - Do not get lured with impossible offers.
    - Handle online transactions carefully.
    - Make payments through secured connections.
    - Use ad blocker.
    - Beware of cookies. Cookies are small text files on the computer storing small pieces of information related to online habits of the user.
    - Always bookmark important sites.
    - Choose for private browsing through Incognito browsing, Proxy and Virtual Private Network (VPN).
  - **Cyber law** is the part of the legal system that deals with cyberspace, internet and legal issues. It covers a broad area, like freedom of expression, access to and utilization of internet, and online security or online privacy. It is commonly known as the law of the web.
  - The Information Technology Act, 2000 (also known as ITA-2000, or the IT Act) is an Act of the Indian Parliament (No. 21 of 2000) notified on 17 October, 2000. It is the primary law in India dealing with cybercrime and electronic commerce.
  - **Safely Accessing Websites**
    - While accessing any website we should always be aware that it's safe and it does not contain any harmful content in it like virus and other malicious things always try to access websites which has https:// in it rather http:// here "s" means secure.
    - A Threat is a potential violation of security. When a threat is actually executed, it becomes attack. Those who executes such actions, or cause them to be executed are called attackers. They are also known as Malware that harms your computer.
  - Some common threats the average computer user faces everyday are
    - **Computer Viruses:** Computer Virus are malicious code/programs that cause damage to data and files on a system. Virus can attack any part of a computer's software such as boot sector, operating system, system areas, files, applications-program-macro, etc.  
Two other similar programs also cause virus like effects. These are:
      - (1) **Worms:** a worm is a self-replicating program which eats up the entire disk space or memory. A worm keeps on creating its copies until all the disk space or memory is filled.
      - (2) **Trojans Horse:** A destructive program that masquerades as a benign application. Unlike viruses, Trojan horses do not replicate themselves but they can be just as destructive. One of the most insidious types of Trojan horse is a program that claims to rid your computer of viruses but instead introduces viruses onto your computer.
- Damage caused by VIRUS**
- (1) Viruses damage documents or may even delete specific files. Most of the time a virus is programmed to damage system files.
  - (2) Virus tend to slow down the system by executing itself in the background.
  - (3) Some viruses attach themselves to the contacts in victims address book and then spread through emails.

- **Spyware:** Spyware is a software which is installed on your computer to spy on your activities and report this data to people willing to pay for it. Spyware mostly get installed on your PC without your consent. Roughly 32% computers of the world are infected with some type of Malware.

**A spyware affects the system in following ways.**

- (A) It monitors information about the owners computing habits, record the keystrokes when the user enters some information and thus lead to identity theft.
- (B) It alters PC settings like the browser home page or appearance of desktop.
- (C) It robs off the system or PC speed or internet access efficiency.
  - **Adware:** These are the programs that deliver unwanted ads to your computer. They consume your network bandwidth. Adware is similar to spyware. You have to be very careful while installing a software.
  - **Spamming:** Spamming refers to the sending of bulk-mail by an identified or unidentified source in malicious form, the attacker keeps on sending bulk mail until the mail-server runs out of disk space.
  - Damaging effects of spam
  - Billions of spam messages that get circulated across the internet disrupt mail delivery and degrade system performance and overall productivity.
  - Deleting spam is time consuming and wastes a lot of valuable time.
  - Spam messages may contain fraudulent messages and sometimes act as VIRUS carriers.
  - **PC Intrusion:** Every PC connected to the internet is a potential target for hackers. PC intrusion can occur in any of the following form:
    - (1) **Sweeper Attack:** This is another malicious program used by hackers which sweeps all the data from the system.
    - (2) **Denial of Services:** This eats up all the resources of the system and the system comes to a halt.
    - (3) **Password Guessing:** Most hackers crack or guess passwords of system accounts and gain entry into remote computer systems.
      - **Eavesdropping:** Unauthorized monitoring of other people's communications is called Eavesdropping. Eavesdropping can be carried out through all communication devices and media of today-telephone, emails, messaging, and other internet services. To prevent this, email messages have to be encrypted and digital signature are also effective.
      - **Phishing and Pharming:** In Phishing, an imposter uses an authentic looking e-mail or website to trick recipients into giving out sensitive personal information which may be later used for cyber crimes and frauds. Pharming is an attack in which a hacker attempts to redirect a website's traffic to another, bogus website. Even if the URL is correct, it can still redirect to a fake website.
      - **Cookies:** A Cookie is a small piece of data sent from a website and stored in a user's web browser. Some cookies disappear after user closes his browser while others, known as tracking cookies, remain saved and load the next time user visits the same website. These cookies help track user's browsing sessions and load information faster, but create some security and privacy concerns as well.

**Threats related to cookies are:**

- (A) **Session Data** – On the sites that you visit regularly your username and password information is pulled from a tracking cookie. Chances are that someone may acquire your cookies and find the encryption key to get your passwords.
- (B) Invasion of Privacy by cookies that track patterns of someone's web activity.
- (C) Cookies of your web activity stored in a public computer pose a larger risk of information being accessed without authorisation.

**Solutions to Computer Security Threat**

- The entire computer security is based on a system of actions and safeguards which have been categorized in two ways:
  - (A) **Active Protection:** It includes installing and using an antivirus that has internet security also which shows protection against threats such as viruses, spyware and PC intrusion.
  - (B) **Preventive Measures:** Even though security programs may actively detect and eliminate any threats your PC encounters, you should always help to prevent these issues from ever arising.

## CHAPTER-17

# TECHNOLOGY & SOCIETY AND E-WASTE MANAGEMENT

### Topic-1 E-Waste Management



#### Revision Notes

- e-waste is the discarded computers, electrical and electronic gadgets, entertainment device, mobiles, refrigerators etc.
  - e-waste is created when an electronic product is discarded may be for reuse, recycle, resale or salvage, after the end of its usual life.
  - e-waste also includes scraps (copper, steel, plastic, etc) and the material which is dumped after reuse or recycling operations.
  - It contains toxic pollutants like mercury, lead, cadmium, chromium and plastics.
  - These also contain valuable substances like gold, silver and copper.
  - Its scientific disposal is therefore very important to protect human and environmental health.
- Recycling of e-waste is an upcoming trend due to need to protect environmental and human health.
  - Recycling of e-waste has to be done under a controlled system to prevent pollution and ensure workplace safety and health.
  - Recycling e-waste helps to conserve our earth's precious natural resources.
  - Many materials in e-waste can't be tossed in a landfill.
  - WEEE (Waste Electrical and Electronic Equipment) is e-waste or scrap.
  - Recycling often leads to overseas shipping and dumping of gadgets to third world countries.
  - This leads to health issues in these countries as children there are involved in extraction of precious metals from these e-waste .
  - There is a rise in cybercrimes in these countries as salvaged hard drives can give criminals direct access to some personal data and information.
- **Waste Disposal techniques**
  - (i) Give your e-waste to a certified Recycler.
  - (ii) Sell off or donate your e-waste.
- **e-waste management in India:**
  - Government passed the first law on e-waste management in 2011, based on EPR (Extended Producers Responsibility) which puts the producers responsible for the management for the final stages of the life of its product.
  - This law was amended to set collection targets for the producer.
  - A new arrangement entitled Producer Responsibility Organization (PRO) has been introduced to be authorized or financed collectively by producers to share the responsibility for collection and channelization of e-waste generated from the end-life of products to ensure environmentally sound management of e-waste.

### Topic-2 Technology and Society



#### Revision Notes

- New Technology improves quality of life for human beings and as a side effect makes it complex and has negative effects on society and environment.
- Technology is the reflection of people's imagination on solving existing problems.
- Technology such as computers, smartphones, laptop, etc. have made life easier.



- Exchanging information, making faster decisions, interacting socially, entertainment, financial transaction processing, online shopping, managing homes all have become as easy and fast as blink of an eye.
- These technologies though have made life easier but are posing many societal and environmental hazards.
- The advancement in technology has made many tasks simple and easy as compared to the conventional ways.
- Sending and receiving messages, documents etc has become a matter of few minutes and is much cheaper.
- No more stepping out early from home to wait for a taxi or public transport for going somewhere.
- Technology itself is not harmful for the society but the way we use it leads to the negative impacts of technology.
- **Negative impacts on society**
  - **Resource depletion:** More and more use and fast growing demand of technically advanced gadgets has led to a pressure on mother nature and hence a fast depletion of natural resources.
  - **Increased population:** Technology has helped us live longer by improving health facilities. It has been a negative effect for developing countries.
  - **Increased pollution:** Advancement in technology has led to more and more manufacturing units and hence to environmental pollution.
  - **Lack of social skills:** Frequency of interacting personally has been reduced much thus kids and teenagers are deprived of basic social manner.
  - **Poor sleep Habits:** Endorsing online activities have affected the sleeping pattern of people.
  - **Loneliness/Isolation:** Engaged in our gadgets we get isolated from the world around us even if we are at a crowded place.
  - **Addiction:** Addiction to technology is no less than the drug addiction.
  - **Obesity:** Sitting on social media and dependence on technology for minimal tasks like grocery shopping has led to obesity, Kids don't feel a need to go out and play with friends when they can sit back at home and play online games with their online friends
  - **Depression:** Dependence on technology and less interaction with fellow human beings can lead to depression.
  - **Lack of Privacy:** People are opening up their private space by giving their information on other sites thus giving rise to criminal activities.
  - Overshare on social media has led to the tendency of crossing social boundaries and cyberstalking has become common.
  - Children at much younger age get an exposure to sexual submissions.
  - Sexting has become another major problem.
  - **Lesser attention span:** Constant newsfeed, getting hundred of messages in a minute switching application too frequently has led to our mind being programmed for lesser attention span on a particular task. Hence remembering and recalling are becoming tough tasks.
  - People are losing empathies due to lack of knowledge about social ethics and hence social violence is on an increase.
  - Using earphones, headphones could cause people to reduce their hearing after sometime.
- **Cultural changes induced by technology**
  - Any technological change is the result of the need of a particular culture.
  - The new technology in turn changes the culture and other cultures to which it spreads.
  - Use of online shopping has changed the culture of going out to the market and buying goods.
  - Social media has changed the culture of going to a friend's place to have a chitchat.
  - Chat rooms have changed the culture of road side group discussions.
  - Online Home delivery of foods have changed the culture of going out for a dinner or lunch with family and also the culture of home cooked food.
  - Digital Natives is applied to the people born after 1980-1990 the time when social digital technologies came online in west. These people have access to the network digital technologies.
  - The use of digital technology has changed the way they think or work. Apps have changed a person to a tailor-made self where they are distracted from the inner self.
  - The boundary between private and public spheres has diminished and people show how they appear to live rather than how they really live.
  - Interculturality is another aspect where people are aware of a global outlook of an issue but lack a deeper understanding due to a poor cultural background.
  - Cyberculture is the term coined to mean the set of material and intellectual techniques, practices, attitudes ways of thinking and values that are expressed and developed in cyberspace.
- **Gender and Disability issues while Teaching and Using Computers**
  - **Gender issues:** It has been commonly observed that girls are under represented in Computer Science Studies be it computer science major subject at high school or at college level. There are far less girls than boys in computer science section.



- The factors behind this are:
- 1. **Preconceived notions:** Notions like boys are better at technical things, girls are good at humanities, arts etc.; girls must take up a career keeping in mind that they have to raise a family; Teaching is the best option for girls as it gives you half day off and ample number of holidays so that you can take care of your family; have their impact on decision making of girls while taking up subjects. Also parents play a major role while deciding the subjects.
- 2. **Lack of interest:** During primitive years children often play games on the computers and smart phones. Most of the games are boys centric that increase the interest of boys in computers.
- 3. **Lack of motivation:** Girls are pressurized to choose a career option which will give them work life balance because they have to play an important family role in future. This theory discourages girls from taking up technical subjects like computer science.
- 4. **Lack of role models:** Girls these days see less role models in the field of computer science. In every field such as movies, advertisements, scientists etc we see more men than women. All these things influence girls sub-psychology and they infer that computer science is for boys and not to take that subject.
- 5. **Lack of encouragement in class:** As there are lesser number of girls in a class, the teachers for most work – assignments end choosing more boys. Also some teachers pin point on their roles in the society. All this may play as hindrance and girls do not develop interest in Computer Science.

#### **Disability Issues**

Various disability issues faced in the teaching/learning computers with regard to the disability are:

- **Unavailability of teaching material/aids:** Students with different disabilities need different types of teaching aids/ material. For instance visually challenged students would want screen readers that could read the digital content to them, hearing impaired students would want more of visual input than oratory, low vision students may need Braille keyboards, braille monitors and printers along with screen readers for working on computers. Unavailability of such supportive programming aids and software are a big issue that must be handled by the school and the management.
- **Lack of special needs teachers:** For different types of special needs, students require special needs teachers. For example: a hearing impaired student needs a teacher who is able to converse in sign language and would be able to convey and explain study material. There should be teachers who know what type of hardware and software tool can be used for the differently able students as per their specific needs. For example braille keyboards, monitors and printers, synthetic speech generators etc, software assistants like Google assistant, Microsoft Narrator and Cortana, specialized editors for visually impaired students for typing programs etc.
- **Lack of supporting curriculum:** Curriculum should be designed while keeping focus on inclusive education. Softwares and programs should be so used so that disabled students can easily work on that. For example office software based curriculum can be easily implemented as it provides accessibility features and all types of students can easily work on it.

