Module 1: Introduction to C

Computer: A computer in simple terms can be defined as an electronic device that is designed to accept data, perform the required mathematical and logical operations at high speed and output the result.

Characteristics of computer:

1. Speed:

- Computers can perform millions of operations per second.
- The speed of computers is usually given in nanoseconds and 1 picoseconds (1 nanoseconds=1*10^-9 and 1 Pico seconds=1*10^-12 seconds)

2. Accuracy:

A computer is a very fast, reliable and robust electronic device. It always gives accurate results, provided the correct data and set of instructions are input to it. If the input is wrong, then the output will also be erroneous, In computer terminology, this is known as **garbage-in garbage-out(GIGO).**

3. Automation:

Computers are automatable devices that can perform a task without any user intervention. The user just needs to assign the task to the computer, after which it automatically controls different devices attached to it and executes the program instruction.

4. Diligence:

Unlike humans, a computer never gets tired of a repetitive task. It can continually work for hours without creating errors.

5. Versatile:

Versatility is the quality of being flexible .Today, computers are used in our daily life in different fields. For example they are used as personal computers (PCs)for home use, for business-oriented tasks ,weather forecasting, space exploration, teaching, railways, banking, medicine, and so on, indicating that computers can perform different tasks simultaneously.

6. Memory:

similar to humans, computers also have memory .While the internal memory of computers is very expensive and limited in size the secondary storage is cheaper and of bigger capacity.

7. No IQ:

Although the trend today is to make computers intelligent by including artificial intelligence (AI) in them, they still do not have any decision-making abilities of their own. That is, their IQ level is zero. They need guidance to perform various tasks.

8. Economical:

Using computers also reduce man power requirements and leads to an elegant and efficient way

of performing various tasks. Hence, computers save time, energy, and money. When compared to other systems, computers can do more work in lesser time.

STORED PROGRAM CONCEPT

All digital computers are based on the principle of stored program concept, which was introduced by Sir John von Neumann in the late 1940s. "**A stored program architecture is a fundamental computer architecture wherein the computer executes the instructions that are stored in its memory".** The following are the key characteristic features of this concept:

• Before any data is processed, instructions are read into memory.

• Instructions are stored in the computer's memory form execution.

• Instructions are stored in binary form (using binary numbers—only 0s and 1s).

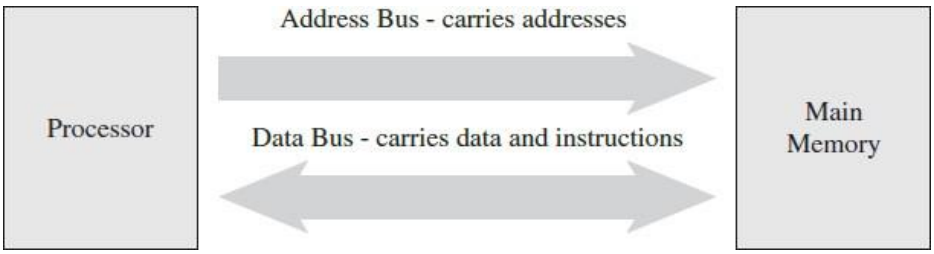
• Processing starts with the first instruction in the program, which is copied into a control unit circuit. The control unit executes the instructions.

• Instructions written by the users are performed sequentially until there is a break in the current flow.

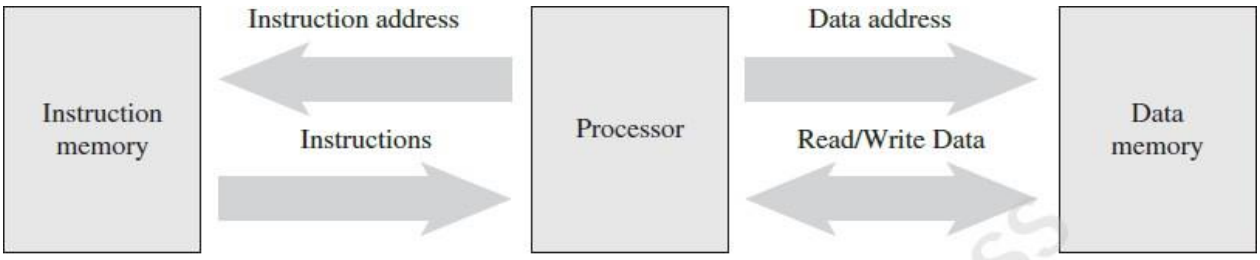
• Input/Output and processing operations are performed simultaneously.

Types of Stored Program Computers

1. A computer with a Von Neumann architecture stores data and instructions in the same memory.



2. Later Harvard University proposed a stored program concept in which h there was a separate memory to store data and instructions.



Computer generations :

Generation in computer terminology is change in technology a computer is being used. There is five generation of computers.

First generation of computer (1940-1956): vacuum tubes:

- This generation computer is used thousands of vacuum tubes for circuitry for CPU (Central processing unit)
- Memory requirement were met by magnetic drums.
- Took lot of space consumed enormous amount of power and generated lot of heat and also was very expensive.
- These computers used machine language.
- Some computers of this generation were: ENIAC, EDVAC, UNIVAC etc..
- Program input was provided by punch cards and outputs were produced on paper.

Second generation of computer (1956-1963) Transistor:

- Transistors were used that were cheaper, consumed less power, more compact in size, more reliable and faster.
- Magnetic cores were used as the primary memory and magnetic tape and magnetic disks as secondary storage devices.
- These computers used assembly language (programmer could specify words and symbols as instructions instead of 0's and 1's).
- Some computers of this generation were: IBM 1620, IBM7094.

Third generation computer (1964-1971) Integrated circuits:

- A single IC has many transistors, resistors and capacitors along with the associated circuitry.
- This development made computers smaller in size, reliable and efficient.
- In this generation remote processing, times sharing multi-programming operating system were used.
- High level language (FORTRAN, COBOL, PASCAL, BASIC, ALGOL) were used during this generation.
- Keyboard and monitor were used to interact with user.
- Memory capacity increased substantially by the use of hard disks for secondary memory.
- Some computers of this generation were: IBM360 series, Honeywell-6000 series, and personal data processor.

Fourth generation computer (1971-1980) Microprocessor:

- Used very large scale integrated (VLSI) circuits having about 5000 transistors and other circuit elements with their associated circuits on a single chip.

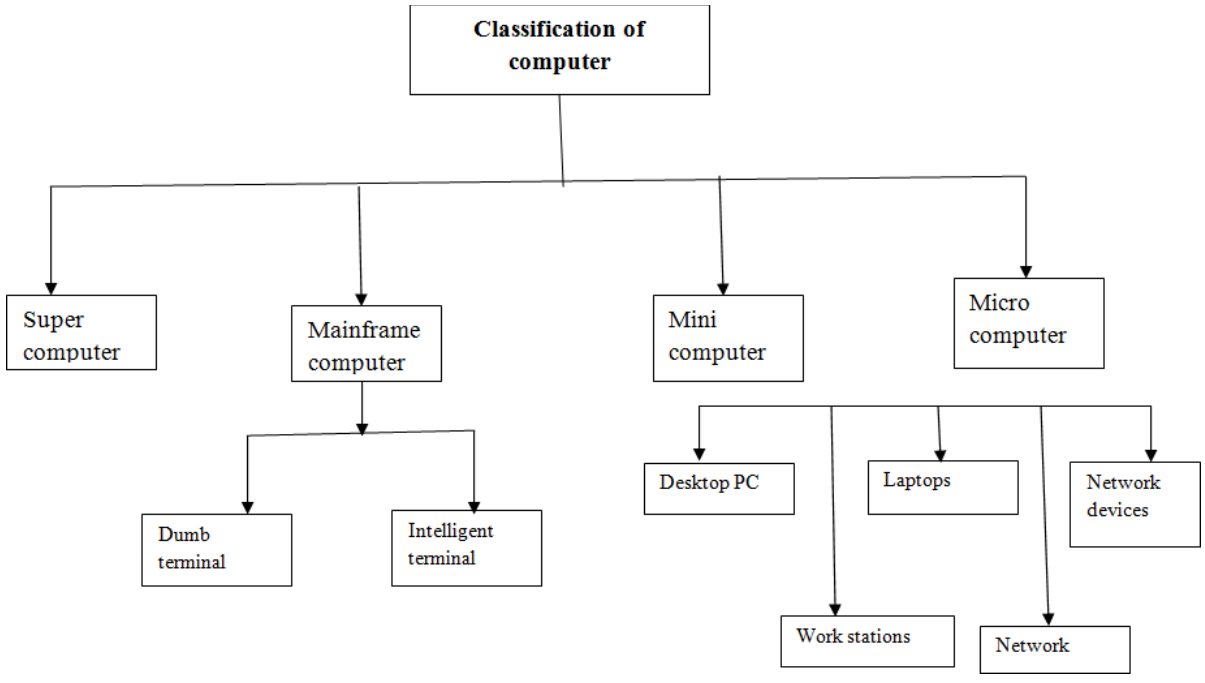
- The microprocessor launched the fourth generation of computer, with thousands of integrated circuits built onto a single silicon chip.
- Computers became more powerful, compact, reliable and affordable.
- As a result, it gave rise to personal computer revolution, laptops and smart phones offering gigabytes of memory compared to few megabytes.
- In this generation, time sharing, real time networks, distributed operating system were used the high level languages like C,C++,DBASE etc..Were used in this generation
- Fourth generation computers also saw the development of graphical user interface, the mouse, and handled devices.

Fifth generation computer(1980-till date)Artificial intelligence:

- With ULS (ultra large scale integration) technology, microprocessor chips having ten million electronic components were manufactured.
- This generation is based on parallel processing hardware and AI.
- AI is an emerging branch in computer science which interprets the means and method of making computers think like human beings.
- All the high-level languages like C and C++,java ,.Net etc..are used in this generation .
- Neural networks and experts systems have to be developed.

Classification of computer:

Computers are broadly classified into 4 types based on speed, amount of data that they can hold and price.



1. **Super computer:**

- The super computer is the fastest, most powerful and most expensive computer.
- Supercomputers were first developed in the 1980's to process large amounts of data and to solve complex scientific problems.
- A single supercomputer can support thousands of users at the same time.
- Computers are mainly used for weather forcasting, nuclear energy research, aircraft design, automotive design, online banking, controlling industrial units, etc.

2. **Main frame computers:**

- Mainframe is very large in size and is an expensive computer capable of supporting hundreds or thousands of users simultaneously.
- The processor speed is measured in MIPS (million instructions per seconds).
- Used for online transactions.
- Capability to handle large amount of data makes mainframe suitable for use in government, banks and financial institutions.
- There are basically two types of terminals that can be used with mainframe systems are:

i. **Dumb terminal:-**Dumb terminals consist of only a monitor and a keyboard (or mouse).They do not have their own CPU, memory and use the mainframe systems CPU and storage devices.

ii. **Intelligent terminals:-**Intelligent terminals have their own processor and thus can perform some processingoperations..They do not have their own storage space.

3. **Mini computer:**

- It is multi-user computer system, capable of supporting hundreds of users simultaneously.
- Minicomputers are smaller, cheaper and slower than mainframes. They are called minicomputers because they were the smallest computer of their times. Also know as midrange computers, the capabilities of minicomputers fall between mainframe and PCs.
- The first minicomputer was introduced by digital equipment corporation (DEC) in the mid 1960.
- The minicomputers are used in small organization or a department of a large one.

4. **Micro computers:**

- Microcomputers commonly known as PCs are very small and cheap.
- The microcomputer or PC is introduced by Apple and endorsed by IBM. This is a single-user machine powered by a single-chip microprocessor.
- They are very powerful machines havinggigabytes of memory.
- They are both used in standalone mode and in a network.
- A microcomputer takes the form of desktop, notebook (laptop) or a notebook (smaller

laptop). PCs today are powered by 3 types of OS – windows (7, 8 or 10), Mac OS X (Apple) and Linux.

- They are used for engineering and scientific applications and for software development.
- PCs can be classified into the following categories.
    - Desktop
    - Laptops
    - Workstations

APPLICATIONS OF COMPUTERS

When the first computers were developed, they were used only in the fi elds of mathematics and science. Today, computers are widely used in fi elds such as engineering, health care, banking, education, etc.

1) **Word processing** -Word processing software enables users to read and write documents. Users can also add images, tables, and graphs for illustrating a concept. The software automatically corrects spelling mistakes and includes copy–paste features.
2) **Internet** The Internet is a network of networks that connects computers all over the world. It gives the user access to an enormous amount of information, much more than available in any library.
3) **Digital video or audio composition** Computers make audio or video composition and editing very simple.
4) **Desktop publishing** software enables us to create page layouts for entire books.
5) **e-Business** or electronic business is the process of conducting business via the Internet. This may include buying and selling of goods and services using computers and the Internet.
6) **Business-to-consumer** or B2C In this form of electronic commerce, business companies deploy their websites on the Internet to sell their products and services to the customers.
7) **Business-to-business** or B2B This type of electronic commerce involves business transactions performed between business partners.
8) **Consumer-to-consumer** or C2C This type of electronic commerce enables customers to carry business transactions among themselves.
9) **Electronic banking**, also known as cyberbanking or online banking, supports various banking activities conducted from home, a business, or on the road instead of a physical bank location.
10) **Bioinformatics** is the application of computer technology to manage large amount of biological information. Computers are used to collect, store, analyse, and integrate biological and genetic information to facilitate gene-based drug discovery and development.
11) **Health care-** storing records, surgical procedure, better diagnosis and treatment.
12) **Geographic Information System and Remote Sensing**- A geographic information system (GIS) is a computer based tool for mapping and analysing earth's features. Remote sensing is the science of taking measurements of the earth using sensors on airplanes or satellites.
13) **Meteorology** – weather forecasting, agriculture, nuclear and maritime meteorology.
14) **Legal –** computers are used by lawyers.
15) **Multimedia and animation**
16) **Retail business –** in stores.
17) **Travel and tourism**
18) **Education**
19) **Astronomy**
20) **Industry and engineering**

Basic organization of computer

A computer is an electronic device that basically performs five major operations:

- Accepting data or instruction
- Storing data
- Processing data
- Displaying data
- Controlling and coordinating all operations inside a computer.

Input:

- This is the process of entering data instructions into the computer system.
- The data and instructions can be entered by using different input devices such as keyboard, mouse, scanner, trackball, etc...
- Note that the computers understand binary language, which consist of only two symbols (0 and 1), so it is the responsibility of the input devices to convert the input data into binary codes.

Storage:

Storage is the process of saving data and instructions permanently in the computer so that they can be used for processing.

A computer has two types of storage areas:

1. **Primary storage:**
    - Primary storage also known as the main memory is the storage area that is directly accessible by the CPU at very high speed.
    - Drawback of main memory is that it is volatile in nature that is as soon as the computer is switched off, the information stored gets erased.
    - RAM and ROM are the examples of primary storage

    RAM:

    - It stands for random access memory.
    - It is a temporary memory that means it is used to store information that is used immediately.
    - Once the computer is turned off the data will be deleted. With the help of RAM computers can perform multiple tasks like loading applications, browsing the web tasks like loading applications, browsing the web, editing a spreadsheet etc…
    - There are 2 types:

    1.S-RAM (static RAM): Volatile, stores until system is on

2. DRAM(Dynamic RAM): Store binary bits in the form of electrical charges.

ROM:

- It stands for read only memory the data stored in these devices are non-volatile i.e. oncethe data stored in the memory cannot be modified or deleted.
- The memory form which will only read but cannot write.
- The information is stored permanently.There are 3 types
  1. PROM (programmable ROM) :Data cannot be altered
  2. EPROM(Erasable Programmable ROM): Possible to erase the information
  3. EEPROM (electrically Erasable ROM): non-volatile memory used in computers, usually integrated in microcontroller

2. Secondary storage:

- Also known as auxiliary memory this is just the opposite of primary memory. It basically overcomes all the drawbacks of the primary storage area.
- It is cheaper, non-volatile and used to permanently store data and programs of those jobsthat are not being currently executed by the CPU.

Difference between primary and secondary storage devices:

Primary storage device/primary memory	**Secondary storage device/secondary memory**
It is a temporary (volatile)memory	It is permanent (Non-volatile)memory
Limited storage space	Huge storage space
There will be loss of data when power goes off	There is no loss of data when power goes off
Primary memory is known as main memory	Secondary memory is known as additional or back memory
Data is directly accessed by the processing unit	Data cannot be accessed directly by the processor. It is first copied from secondary memory to primary memory. Only then CPU can access it.
Primary memory is costlier than secondary memory	Secondary memory is cheaper than primary memory.

Processing:

- The process of performing operations on the data as per the instructions specified by the user(programs) is called processing.
- Data and instructions are taken from the primary memory and transferred to the arithmetic and logical unit, which performs all sorts of calculations.

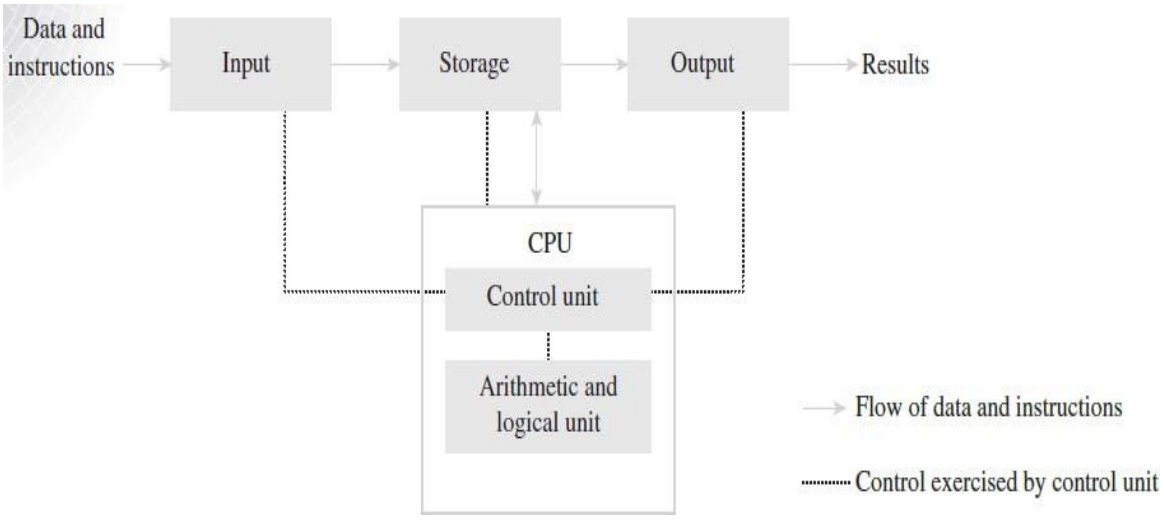
- The intermediate results of processing may be stored in the main memory as they be required again.
- When processing completes the final result is then transferred to the main memory .Hence the data may move from main memory to the ALU multiple times before the processing is over.

Output:

- Output is the processing of giving the result of data processing to the outside world.
- The results are given through output devices such as monitor, printer etc...
- The output devices therefore convert the results available in binary codes into a human-readable language before displaying it to the user.

Control:

The control unit is the central nervous system of the entire computer system. It manages and controls all the components of the computer system. It is the CU that decides the manner in which instructions will be executed and operations performed



MOTHERBOARD

The motherboard, also known as the mainboard or the parent board is the primary component of a computer. It is used to connect all the components of the computer. The motherboard is a printed circuit that has connectors for expansion cards, memory modules, the processor, etc.

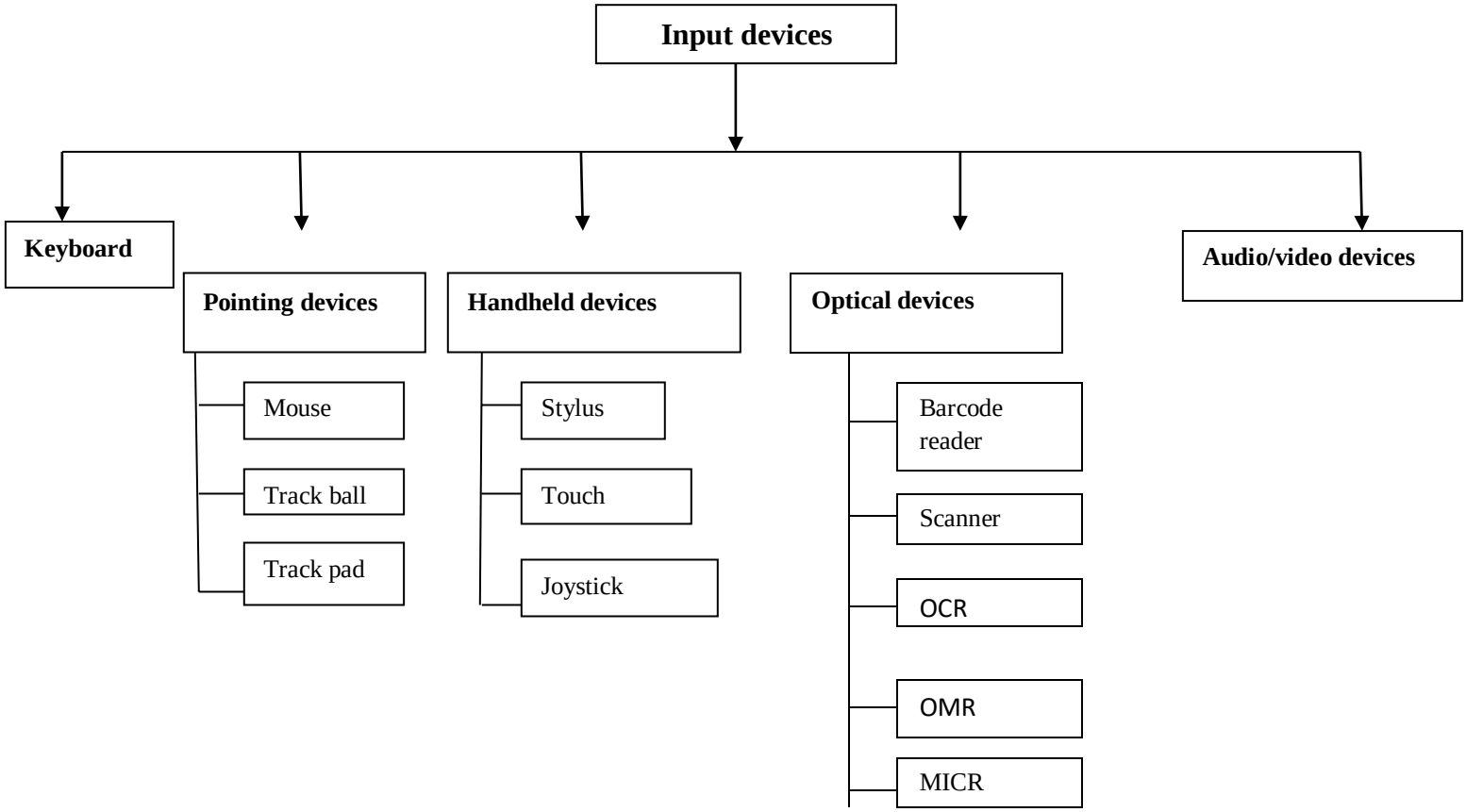
Characteristics of a Motherboard

A motherboard can be classifi ed depending on the following characteristics:

• Form factor
• Chipset
• Type of processor socket used
• Input–Output connectors

Input and output devices:

An input device is used to feed data and instructions into the computer. In the absence of an input device, a computer would have only been a display device.



Keyboard:

- The keyboard is the main input device for computers. Computer keyboards look very similar to the keyboards of typewriters with some additional keys.
- Most keyboards have between 80 and 110 keys.
- The layout of a keyboard is known as **QWERTY** for its first six letters.
- Types of keys:
    a. Typing keys – includes letters of alphabets.
    b. Numeric keys – set of keys arranged like a calculator with numbers and symbols (+,-,*,/,.,).
    c. Function keys – used by applications and OS to input specific commands.
    d. Control keys – they are used to handle control of cursor and the screen. Four arrow keys , Home, End, Insert, Delete, Page up and page Down keys, Ctrl, Esc, Alt, Pause, PrintScreen , Windows, Start Keys are some of the control keys.

Key board has circuitry called *key matrix*, which will be completed when the key is pressed.

Pointing devices:

- Pointing input devices enable the users to easily control the movement of the pointer to select items on a display screen to select commands menu, to draw graph, etc...
- Some examples of pointing devices include mouse, track ball, light pen joystick and touch pad

a. Mouse:

- The mouse is an input device that was invented by Douglas Engelbart in 1963.
- It is the key input device used in a graphical user interface
- It can be used to handle the pointer easily on the screen to perform various functions such as opening a program or file.
- The mouse has two buttons and a scroll wheel. It can be held in the hand and easily moved without lifting, along a hard flat surface to move the cursor to the desired location-up, down, left, right.
- Once the mouse is placed at appropriate postion,the user may perform the following operation:
    - **Point:** Placing the mouse pointer over the word or the object on the screen by moving the mouse on the desk is termed as pointing.
    - **Click:** pressing either the left or the right button of the mouse is known as clicking.
    - **Drag:** Pointing to a desired location while pressing the left button.

The most popular Mouse types are:

i. **Mechanical mouse:-** This type of mouse has a rubber or metal ball at its bottom and an electronic circuitcontaining sensors.

ii. **Optical mouse:-** The movement of the mouse is detected using laser technology, by using optical sensors.

iii. **Cordless mouse:-** is not connected to the computer. The movement of the mouse is detected using radio waves or infrared light.

b. Track ball:

- A trackball is a pointing device that is used to control the position of the cursor on the screen. It is usually used in notebook computers, where it is placed on the keyboard.

c. Touchpad:

- A touchpad (track pad) is a small, flat, rectangular stationary pointing device with a sensitive surface of 1.5-2 square inches.
- The user has to slide his or her fingertips across the surface of the pad to point to a specific object on the screen.
- The surface translates the motion and position of the users fingers to relative position on the screen.

- There are also button around the edge of the pad that work like mouse buttons.
- Touch pad are widely used in laptops and are in built on the laptop keyboard.

Handheld device:

Joystick: A joystick is a cursor control device widely used in computer games and computer aided design(CAD)/computer aided manufacturing (CAM) applications.

Stylus:- A stylus is a pen-shaped input device used to enter information or write on the touchscreen of a handheld device.

Touch screen:

- A touch screen is a display screen that can identify the occurrence and position of a touch inside the display region.
- The user can touch the screen either by using a finger or a stylus.
- The touch screen facilitates the users to interact with what is displayed on the screen in a straightforward manner, rather than in an indirect way by using a mouse or a touchpad.

Optical devices:

- Optical devices also known as data-scanning devices use light as a source of input for detecting or recognizing different objects such as characters, marks, codes and images.
- The optical device converts these objects into digital data and sends it to the computer for further processing

Barcode reader:

- A barcode reader is a handheld input device that ais used to capture and read information stored in a barcode
- It consists of a scanner a decoder and a cable used to connect the reader to a computer.
- The function of the barcode reader is to capture and translate the barcode into numerical and/or alphabets.
- It is connected to computer for further processing of the captured information.

Image scanner:

A scanner is a device that captures images, printed text and handwriting from different sources such as photographic prints, posters and magazines and converts them into digital images for editing and displaying on computer.

Optical character recognition:

ptical character recognition is the process of converting printed materials into text or word processing files that can be easily edited and stored.

The steps involved in OCR include:

1. Scanning the text character by character
2. Analyzing the scanned image to translate the character images into character codes.

Optical mark reader:

- Optical mark recognition is the process of electrically extracting data from marked fields such as checkboxes and fill in fields on printed forms.
- The optical mark reader is fed with an OMR sheet that has pen or pencil marks is pre-defined positions to indicate each selected response marks and space and stores the interpreted data in a computer for storage analysis and reporting

Magnetic ink character reader:

- Magnetic ink character reader is used to verify the legitimacy of paper documents especially bank checks.
- It consists of magnetic ink printed characters that can be recognized by high speed magnetic recognition devices.

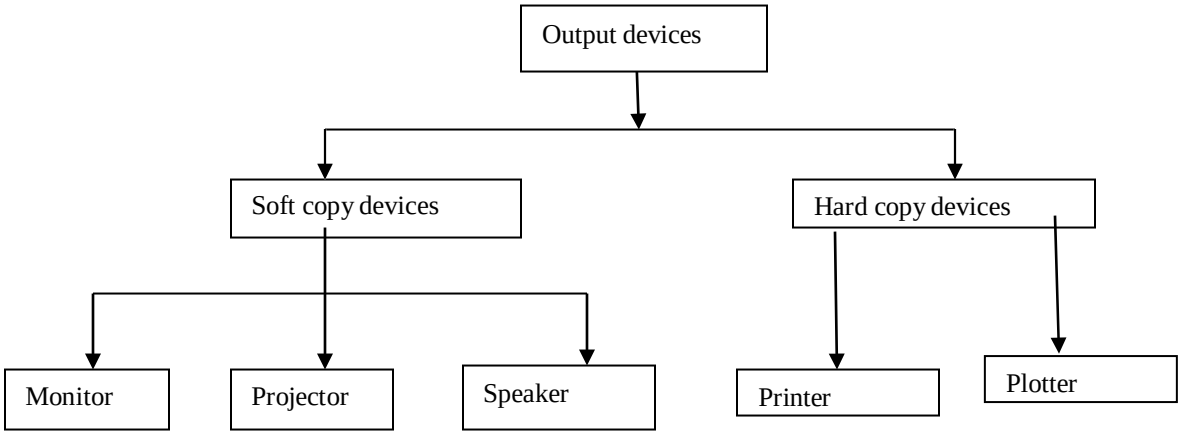
Audiovisual input devices:Audio devices:

- Audio devices are used to either capture or create sound
- They enable computers to accept music, speech or sound effects for recording and /or editing.
- Microphones and CD players are examples of two widely used audio input devices.
- A microphone feeds audio input to the compter.however the audio input must be converted into digital data before being stored in the computer.

Video input devices:

- Video input devices are used to capture video from the outside world into the computer.
- **Digital camera** and webcam are the popular examples of video input device.
- A digital camera is a handheld and easily portable device used to capture images or videos.
- The digital cameras digitalize images or video and store them on a memory card.
- **Web cameras** too capture videos that can be transferred via the internet in real-time.
- Web cameras are widely used for video conferencing

Output devices:



Monitor: The monitor is an integral part of computer which displays both text and graphics. The performance is measured in terms of image quality, resolution, energy consumption.

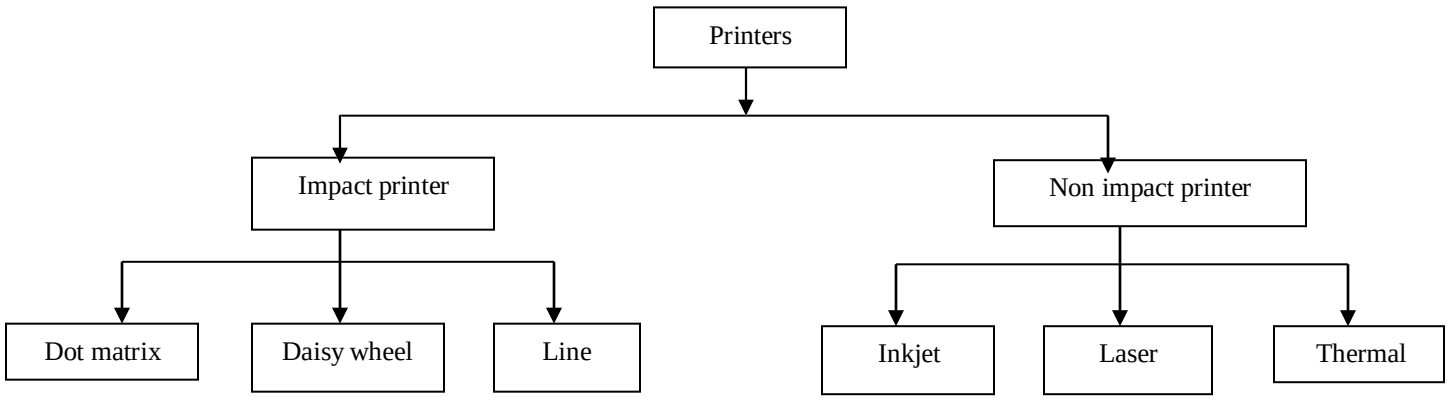
CRT (Cathode Ray Tube) Monitors:

- Cathode ray tube is a specialized vacuum tube in which images are produced whenan electron beam strikes a phosphorescent surface.
- The smaller the pixel, the better image clarity or resolution.
- A finite number of characters can be displayed on screen at once.
- Most screens are capable of displaying 80 characters of data horizontally and 25 lines vertically.
- It is large in size and it consumes very high power.
- They usually have resolution of 640*840 pixels. They are large, heavy, energyefficient and produce lot of heat.

LCD (Liquid Crystal Display) Monitors:

- Flat panel display is a thin screen display found on all portable computers and is thenew standard for desktop computers.
- Unlike monitors, flat panel displays use liquid-crystal display(LCD) or light- emitting diode(LED)technology to make them much lighter and thinner compared to a traditional monitor.
- The image is formed by applying voltage on crystals.
- The backlight is provided by fluorescent light. They consume less power, generate less heat and have increased life span.

Printer: Printer is an output device that prints any data, report, document, picture, diagrametc. Printers are classified into:



Plotters: The plotter can make drawings. It uses one or more automated pens. The commands are taken from special file called vector graphic files.

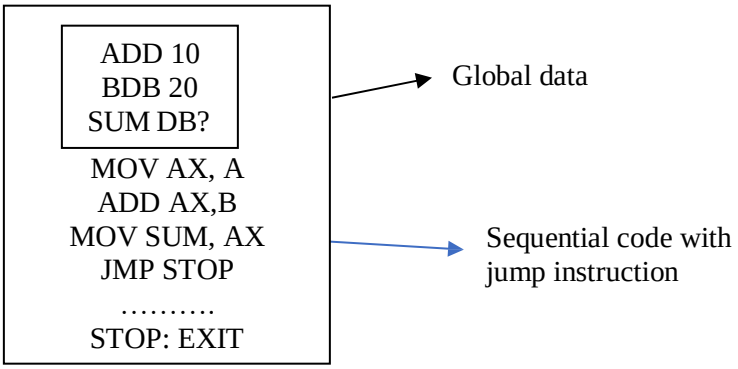
Speakers: The speaker is an output device which produces the voice or sound waves.

Projector: The projector is the by-pass device for the monitor where it produces larger images or pictures

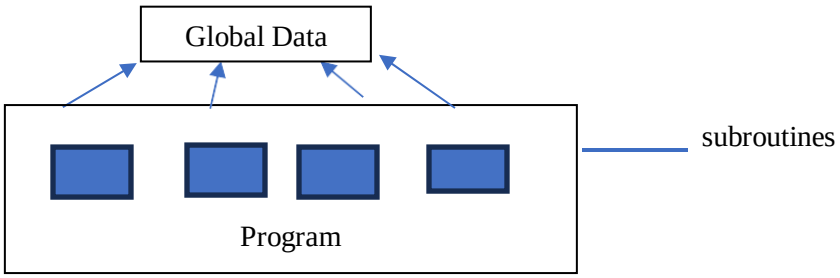
Designing efficient programs

Programming paradigm is a fundamental style of programming the define how the structure and basic elements of computer program will be built. These paradigms in sequence of their application can be classified as follows

1. **Monolithic program**- programs written using monolithic programming languages consist of global data and sequential code. The Global data can be accessed and modified from any part of the program. A sequential code is one in which instructions are executed in a specific order. To change the sequence of execution, **jump or goto statements.** It will have just **one program module.** The size of the program will be large as there is no concept of subroutine and difficult to debug.



2. **Procedural programming**- in procedural languages program is divided into sub routine that can access Global data. The subroutines perform well defined task and hence avoids the repetition of code and reduces size of program. A subroutine can take service of another subroutine. The sequence of the instruction can be changed by **jump, goto and call** instructions.



Example FORTRAN and cobol are two popular procedural programming language.

Advantages

a) The only goal is to write correct programs.
b) Programs are easier to write compared to Monolithic programming.

Disadvantages

a) No concept of reusability.
b) Requires more time and effort to write programs.
c) Programs are difficult to maintain.
d) Global data are shared and may get altered.

3. **Structured programming**

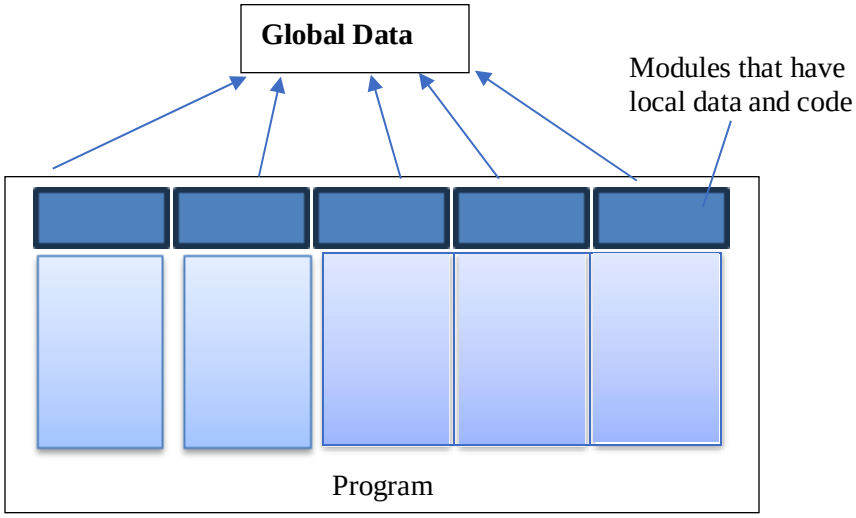
It is also referred to as modular programming. It will be used basically in large programs that require large development team to develop different parts of the same program. Structure programming employs top down approach in which overall program structure is broken down into separate modules. This allow the program to be loaded efficiently into the memory and also be reused in other programs. Modules are coded separately, tested individually and then integrated to form overall program structure. It is easier to debug and understand the program.

Advantages

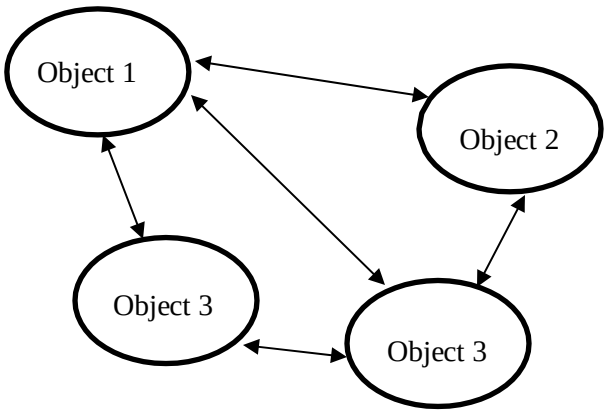
a) Easy to understand and change.
b) Many programmer can work on different modules of large program.
c) Can be written in less time.
d) Module or procedures written once can be reused in other program.
e) Easy to debug.

Disadvantages

a) Not data centered
b) Global data is shared and gets modified.



4. Object oriented programming (OOP)
   This pe programming paradigm helps to develop maintainable programs. It over comes the disadvantages of the previous para paradigms. It treats data as a critical element in the program development. The Monolithic structure and procedural programming paradigms are task based as the focus on action the program should accomplish. In OOP paradigm all the relevant data and task are group together in entities known as objects.
   Features :
   a) Programs are data centered.
   b) Programs are divided in terms of objects and not procedures
   c) Functions that operate on data are tied together with data.
   d) Data is hidden and not accessible by external function.
   e) New data and functions can be easily added as and when they are required.
   f) Follows a bottom up approach for problem solving.



Design and implementation of efficient programs

The design and development of correct, efficient and maintainable programs developer has to follow a development process. The entire software development process is divided into number of phases. The output of one face provides the input for the subsequent phase. The phases are called software development life cycle(SDLC).

Requirements analysis

In this space the uses expectations are gathered to know why the program or software has to be built. All the gather requirements are analysed the objective of the software product. Every identify requirement of the users are documented to avoid any doubts.

The functionality capability performance and availability of hardware and software components are analysed in this face.

Design

In design phase plan of action is made before the actual development process can start. The requirements documented in the previous phase acts as input for the design paste. In this space the structure of the program is broken down into modules. The solution for each modules is specified in the form of algorithms or flowcharts.

Implementation

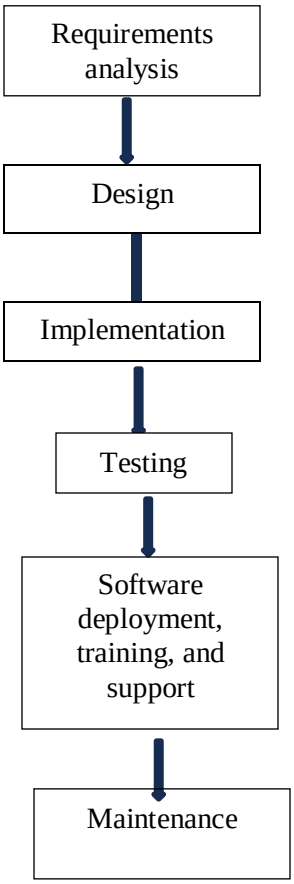
In this phase the designed algorithm are converted into program code using any high level language. This phase is also called as construction of food generation phase.

Testing

In this phase all the modules are tested together to ensure it the overall system works well. The software is tested using a large number of varied inputs also known as test data to ensure with the software is working as expected by the user requirement.

Software deployment training and support

In this phase the software is installed or deployed in the production environment. Maintenance and enhancements are the ongoing activity that are done to cope up with newly discovered problems for your requirements.



Program design tools: Algorithms , Flowcharts , Pseudocodes

We will learn about the different tools which are used to design solutions of a given problem at hand.

1. Algorithm

An algorithm provides a blueprint writing the program to solve the particular problem. It is a step by step procedure to solve a problem. Algorithms are mainly used to achieve software y reuse. Once we have an idea or blueprint of a solution we can print it in high level language such as C, C++, java and so on.

Control structures used in algorithms

And algorithm has a finite number of steps and some steps main involved decision making and repetition. And algorithm may employee 3 control structures namely sequence decision and reputation.

Sequence- each of the algorithm is executive the specific order.

Example Algorithm: To find the of two given numbers:

Step 1: input first number as A
Step 2: input second number as B
Step 3: Set Sum= A+B
Step 4: Print Sum
Step 5: END .

Decision decision statements are used when the outcome of the process depends on some condition. For example if x=y, then print "EQUAL".

The general for of the construct: If condition then process

```
Step 1: input first number as A
Step 2: input second number as B
Step 3: if A=B
          print "EQUAL"
        ELSE
          print "NOTEQUAL"
        [end of if]
Step 4:END
```

Repetition

Reputation involves executive one or more steps for a number of times. It can be executed using construct such as while do while and for loops.

Example

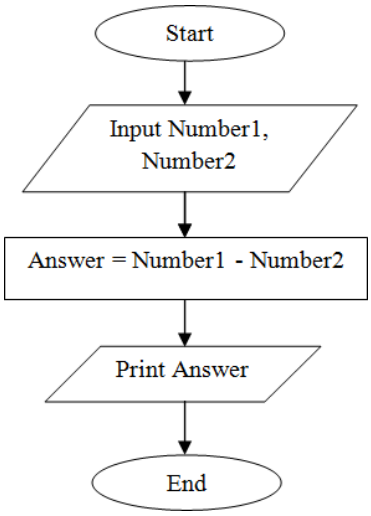
```
Step 1: [INITIALIZE] set I-1, N=10
Step 2: Repeat steps 3 and 4 while I<=N
Step 3: PRINT I
Step 4:SET I+1
        [end of loop]
Step 5:END.
```

2.Flowchart

A flowchart is a graphical or symbol representation of a process. When designing the floor chart, in the process is depicted by different symbols and is associated with a short description. The symbols in the flowchart are linked together with arrow to show the flow of logic in the process.

Symbol	Name	Function
	Start/end	An oval represents a start or end point
	Arrows	A line is a connector that shows relationships between the representative shapes
	Input/Output	A parallelogram represents input or output
	Process	A rectagle represents a process
	Decision	A diamond indicates a decision

A flowchart facilitates communication between programmers and uses. The programmer understands complicated programs easily. They are used for program documentation. The act as a guide or blueprint for the programmers. They also help to read up the programs.
Example



Pseudocodes

Pseudocode is a compact and informal high level description of an aquarium that uses the structural convention of a programming language. It is basically meant for human reading rather than machine reading so it omits that details that are not essential for humans. They consists of short English phrase the explain specific task within a programs algorithm. The soul purpose of the seed of course is to enhance human understand ability of the solution. There are no standards

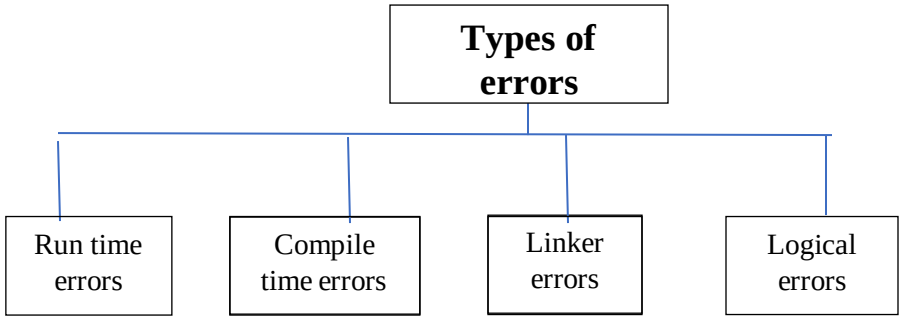
defined for writing pseudo code, because a pseudo code is not an executable program. Flowcharts can be considered as an alternative to the pseudocode but require more space on paper.

Example : pseudocode to add two numbers.

1. Read the first number A.
2. Read the second number B
3. Find the sum =A+B
4. Print sum.

Types of errors

While writing programs very awesome needed errors in our programs. These errors if not removed to either give erroneous output or you not read the compiler compile the program. These errors are broadly classified into four types.



1. **Runtime errors**- occur in the program is being Run executed. Such errors occur in the program performs some illegal operations like
   - Dividing a number by zero
   - Opening a file that already exist
   - Lack of free memory space
   - Finding Where are logarithm of negative numbers

Run time errors may terminate program execution, so the code must be written in such a way that it handles all sort of unexpected errors.

2. **Compile time errors**- occurs at the time of compilation of the program. Such errors may be further classified as
   - Syntax errors- are generated on rules of programming languages.
   - Semantic errors - those which make comply with the programming language but I not meaning to be the compiler.
3. **Logical errors**- errors in the program for that results in unexpected and undesirable output. Such errors are not detected by the compiler and the programmers must check the code and rectify the errors.
4. **Linker errors**- linker is not able to find the function definition for a given prototype.

Testing and debugging approaches

Testing is an activity that is perform to verify the correct behaviour of the program. It is carried out with an intent to find error. There are three types of testing

- **Unit test**- unit testing is applied only on a single unit or module of a program.
- **Integration testing**- two units that have already been tested are combined into a component and the interface between the mistake.
- **system test**- system testing check the entire system.

Debugging- is an activity that includes extension execution testing and core correction. The main aim of debugging is locating errors in the program code. Once the years are located, there isolated and fixed to produce error free code. Different approaches applied for debugging a coding includes

Brute force method

Back tracking method

Cause elimination

Introduction to C programming:

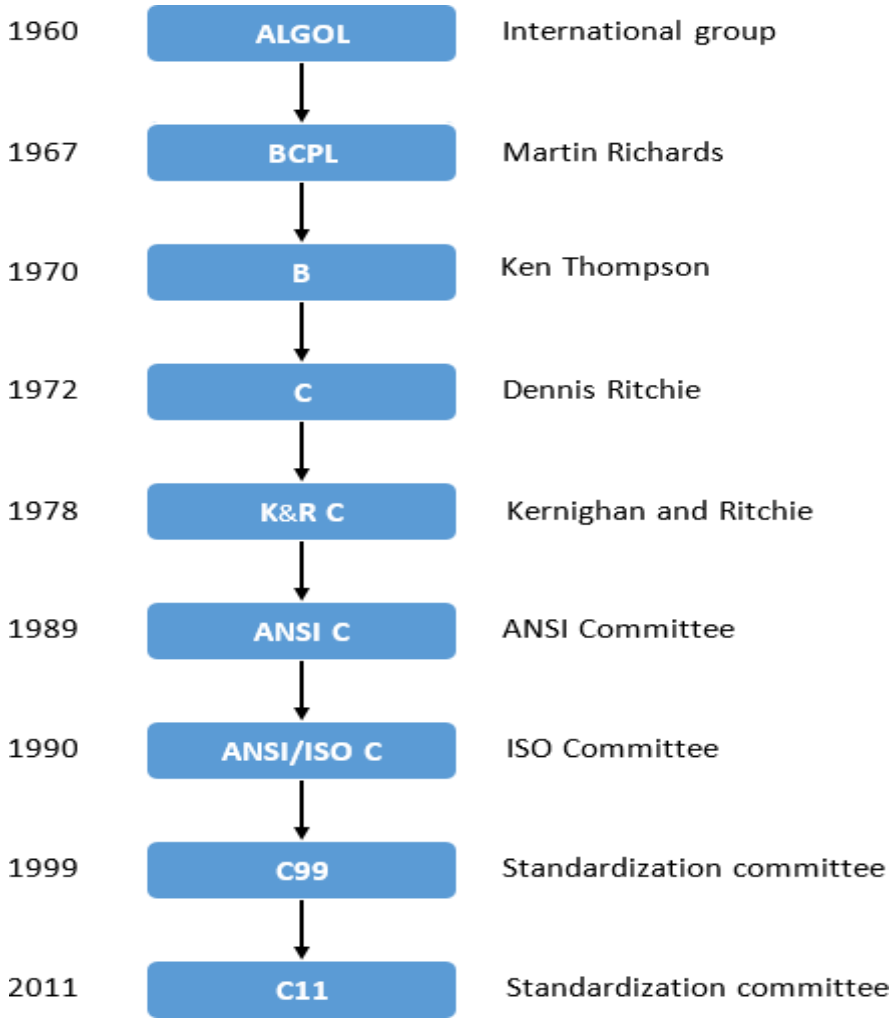
- C is a general purpose, structured programming language. It bridges the gap between machine language and conventional high-level languages. This flexibility allows c language to be used for system programming as well as for applications programming.
- C is a powerful language that provides fast program execution. Its power and fast program execution come from its ability to access low commands similar to assembly language but with high level syntax.

History of C language:

- The c programming language was developed by Dennis Ritchie at AT&T bell laboratories in the early 1970s.He was also known as father of c language.
- Like many other modern language C is derived from ALGOL.ALGOL was not accepted widely in the United States but it was widely used in Europe.
- In 1967 Martin Richard developed a language called BCPL.
- BCPL was basically a type-less language which facilitated the user with direct access of memory.
- 1970 ken Thompson developed a language called B. B was used to develop the first version of UNIX
- C was developed by Dennis Ritchie in 1972 that took concept from ALGOL, BCPL and B.
- In addition to the concepts of these languages, c also supports the concept of data types.
- C was documented and popularized in the book the c programming language by Brain W Kernighan and Dennis Ritchie in 1978.This book was so popular that the language came to be known as “k & RC”.
- In 1983 the American national standards institute started working on defining the standard for C. In December 1989 and came to be known as ANSI C.
- In 1990 the international standard organization adopted the ANSI standard. This version

of C came to be known as C89.

- In 1995 some minor changes were made to C89 the new modified version was known as C95.
- In 1990 C++ and java became popular programming languages among the users so the standardization committee of C felt that a few features of C++/java if added to C would enhance its usefulness. So in 1999 when some significant changes were made to C95, the modified version came to be known as C99.



Features of C language:

Simple robust:

C is a general purpose, structured programming language. It has the simplicity of a high level language as well as the power of low level language. This aspect of c makes it suitable for writing both application programs and system programs. Hence it is an excellent, efficient and general purpose language for most of the application such as mathematical, scientific, business and system software applications.

Small in size and power full:

C languages consist of 32 English words known as keywords. The power c lies in a rich set of built in library functions and variety of operation.

Modularity:

C programs are modular in nature it supports compartmentalization of code and data. This capability makes it very easy for c-programs to share sections of code. We can divide c programs to several modules that will combine in a single module to build the final program.

Extensibility:

The c language is extensible since it allows the users to add their own library function to the library function .section of code or modules can be stored in libraries for re-use them in future.

Portability:

C program written on one computer or operating system can be compiled and run on another computer or operating system with little or no modification.

Efficient and fast:

The compilation and execution of c programs are faster than modern programming languages. This is due to its variety of data type and powerful operators.

Efficient Use of pointers:

C supports efficient use of pointer. Pointers have direct access to memory.

Bit manipulation:

It provides wide variety of bitwise operators to manage data at bit level. We can perform different operations at bit level

STRUCTURE OF 'C' PROGRAM

A C program is composed of preprocessor commands, a global declaration section and one or more functions.

The **processor directive** contains special instructions that indicate how to prepare the program for compilation. One of the most important and commonly used preprocessor commands, include which tells the compiler that some function is needed from the specified header file.

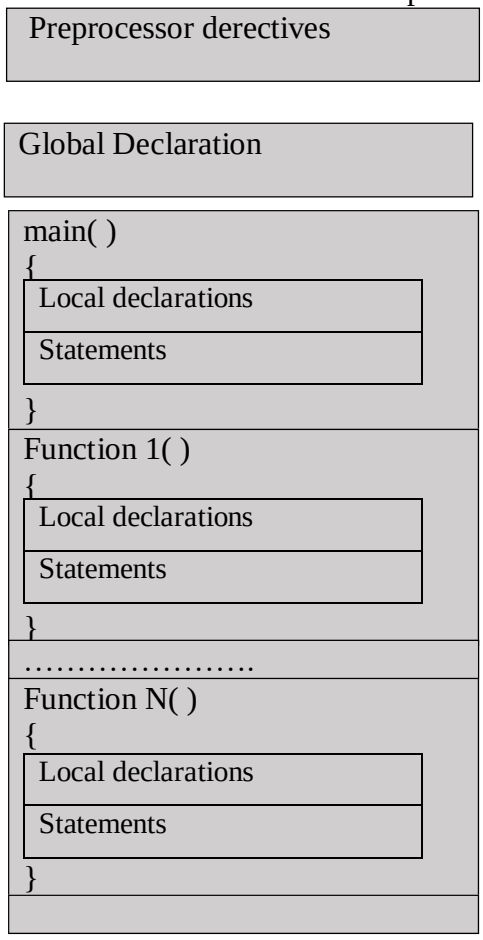
Global declaration includes global data which can be used through out the program.

A C program contains one or more **functions** where a function is defined as the group of two statement that are exhibited together. The statements are written in a logical sequence to perform a specific task.

The **main() function** is the most important function and is a part of every C program. The execution of a C program begins at this function.

All functions are divided into two parts

The **declaration section**- the data are declared within a function are known as local declaration as the data will be visible only within the function.
The **statement section** in the function contains the code to perform specified task.



Writing the first C program

```
#include<stdio.h>
int main()
{
 printf("welcome to the world of C\n");
return 0;
}
```

 # include<stdio.h>
This is a processor command that comes at the first statement in our code. All preprocessor commands start with the symbol hash(#). The # include statement tells the compiler to include standard input/ output library or file in the program. This file has some inbuilt functions which can be used directly in the court by just including this file the program. The standard input output library contains functions to read the data from the keyboard and printing the result on the screen.

int main()

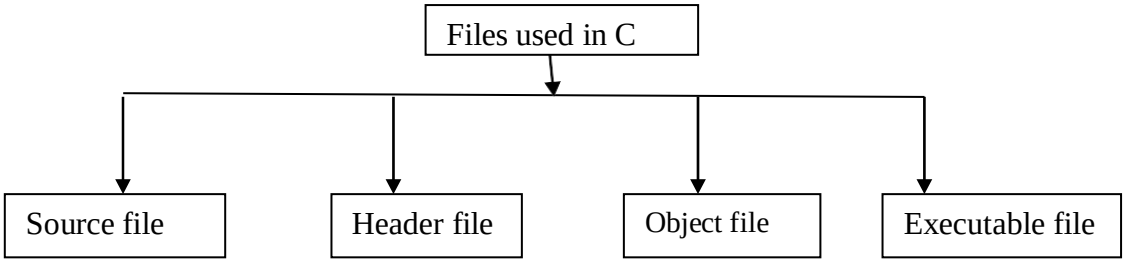
Every C program contains a main() function which is starting point of the program. int is the return value of the main() function. Statement written and integer value to the operating system. To curly brackets{ } are used to group all the related statements in the main function. The statement within the braces from the function body.

The printf function is defined in the stdio.h header file.

Example:

```
//Addition of two numbers
#include<stdio.h>
void main( )
{
        int a, b, sum;
        printf("Enter two numbers\n");
        scanf("%d%d", &a, &b);
        sum = a + b;
        printf("The sum is %d", sum);
}
```

Files used in C program:



Source file:

- The source code file contains the source code of the program.
- The file extension of any C source code file is '.c'. This file contains C source code that define the main function and may be other functions. The main () function is the starting point of execution when you successfully compile and run the program.

Header file:

- When working with large projects, it is often desirable to separate out certain sub routines from the main () of the program.

- There also may be a case that the same subroutine has to be used in different programs. In the latter case one option is to copy the code of the desired subroutine from one program to another. However copying the code is often tedious as well as error-prone and makes maintainability more difficult.

Example:

Stdio.h, stdlib.h, conio.h

Object file:

Object files are generated by the compiler as a result of processing the source code file. They contain compact binary code of the function definition.

Executable file

Linker uses this object file to produce an executable file (.exe file) by combining the object files together. Object file have a '.o' extension.

Compiling and executing a C program:

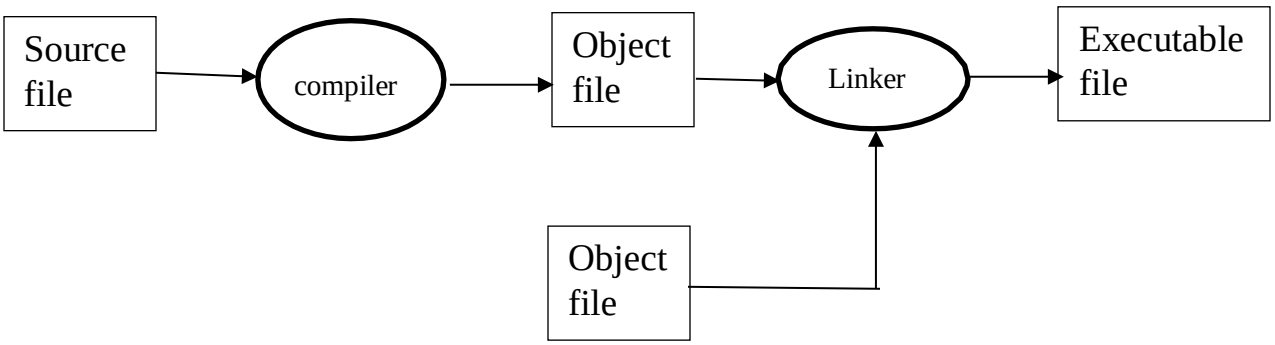
The program development cycle has following four steps:

1. **Writing the source code:**
   The c program is written using vi editor or text editor .Most language compilers have their built-in editors. The program or source code should be written in the syntax provided by that language. In c a source code file has extensions.

2. **Compiling the source code (compilation):**
   The second stage a compilation contains the steps preprocessing, compilation and assembling code. In preprocessing source code is passed to preprocessor. Preprocessor removes comments, expand macros and do expansion of the included files. The source code is compiled using compiler. The compiler checks the syntax errors in the source code. It produce an intermediate compiled output file .obj .
3. **Linking the object code (linking):**
   Linker assembles the entire libraries which are necessary for the program but existing somewhere else. The system uses object code as input and translates it to executable code.

4. **Executing programs(file.exe) :**

   Executable file is ready to run file. Loader loads the executable file into the main/primary memory and run program.



Compilers:

- A compiler is a program that translate the instruction of high level language (source code) into machine language (object language)
- A compiler is a program or set of programs that converts source code written in a high level language to low level language (assembly language or machine language)
- A programming language can have many compilers. For example, GCC,turbo C,Quick C etc.. are different compilers for C programming language.

Need of compiler:

- A computer understands only binary language and executes instructions coded in binary language. It cannot execute a single instruction given in any other form. Therefore we must provide instructions to the computer in binary language.

- Means we must write computer programs entirely in binary language and it is not possible to write the program in binary language.'
- So there was a need of a translator that translates the compiler instructions given in English language to binary language.

Difference between compiler and interpreter:

Compiler	**interpreter**
Compiler takes entire program as input	Interpreter takes single instructions as input
Compiler generates object code	Interpreter is not generates object code.
No source code is required in executable file	Source code is required in executable file
Programming language like C,C++,C# uses compilers	Programming language like python ,ruby uses interpreters.
Compiler translates entire program at a time	Interpreter translates program line by line

Using Comments

Comments in C language are used to provide information about lines of code. It is widely used for documenting code. There are 2 types of comments in the C language.

- Single Line Comments-Single line comments are represented by double slash \\. Let's see an example of a single line comment in C**.**

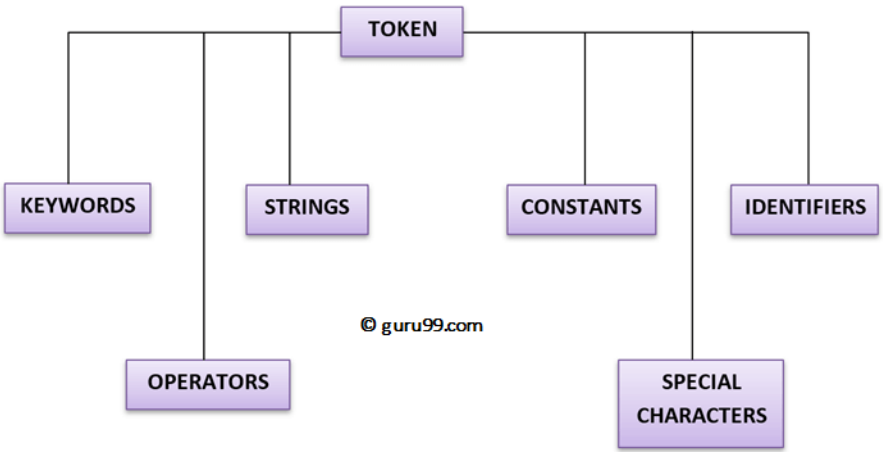
```
#include<stdio.h>
 int main(){
//printing information
  printf("Hello C");
  return 0;  // returns zero value
}
```

- Multi Line Comments- Multi-Line comments are represented by slash asterisk * ... *\. It can occupy many lines of code, but it can't be nested.

```
#include<stdio.h>
int main(){
/*printing information
Multi-Line Comment*/
printf("Hello C");
return 0;
}
```

C Tokens

Tokens are basic building blocks of C. it is the smallest individual unit in a C program. The program is constructed using a combination of these tokens.



Character set:

In the C programming language, the character set refers to a set of all the valid characters that we can use in the source program for forming words, expressions, and numbers.

- English alphabets: uppercase A-Z, and lowercase a-z
- Digits: All numerical digits from 0-9.
- Special characters: . ; : , $ # ^&-+=
- Whitespace: enter ,space, horizontal tab.
- Escape sequence : sequence of characters that doesn't represent itself when used inside string literal or character.

Escape Sequence	Meaning	Elucidation
\n	New line	Used to shift the cursor control to the new line.
\t	Horizontal tab	Used to shift the cursor to a couple of spaces to the right in the same line.
\a	Audible bell	A beep is generated indicating the execution of the program to alert the user.
\r	Carriage Return	Used to position the cursor to the beginning of the current line.
\\	Backslash	Used to display the backslash character.

keywords:

- Keywords are meant for some special purpose called as reserved words.
- They are sequence of characters with fixed meaning.
- User cannot change the meaning of keywords.

auto	double	int	struct
break	else	long	switch
case	enum	register	typedef
char	extern	return	union
continue	for	signed	void
do	if	static	while
default	goto	sizeof	Volatile
const	float	short	Unsigned

Identifiers:

- Identifiers are the names given to the elements of the program such as variable name , name or a function name.
- An identifier is a word consisting of sequence of one or more letters or digits or a special symbol called as _(underscore)

Rules:

- The rules to write an identifier are as follows:
  1. It must contain only alphabets (A to Z, a to z), numbers (0 to 9) and underscore (_).
  2. It must start with alphabet or underscore but not numeric character.
  3. It should not contain any special symbols apart from underscore.
  4. It should not have any space.
  5. Keywords cannot be used as identifiers.
  6. Identifiers are case sensitive.
  7. Maximum length of an identifier is 31 characters.
- **Examples:**

  Valid Identifiers: integer, minimum, sum_total, row1, _cpps

  Invalid Identifiers: float → It is a keyword.

  I am → It has space
  123_Abc → It is starting with number N1 + n2. It contains special symbol (+)

Constants:

Constants are the data values that cannot be change during the execution of a program.

Operators:

It is a symbol that indicates the operation to be performed.

Special symbol:

Special symbols have some special meaning in the c programming language for the compiler and it is used to perform some special function\task in c.

Example: {,}, [,], &,*.

Basic Data types in C

- **Data types:** Data type defines the types of data that is stored in a variable.
- There are 3 types:
  i. **Primary/ Built-in/ Fundamental data type** int , float, double, char, void
  ii. Derived data type → array, structure
  iii. User defines data type → enum , type def.

❖ **Primary/ Built-in/ Fundamental data type**: These are the built in data types available in C. There are5 types. Namely:

1. Integer (int)
2. Floating point (float)
3. Character (char)
4. Double precision floating point (double)
5. Void (void)

1. Integer Type

- It is used to store whole numbers.
- The keyword int is used to declare variable of integer type.
- Int type takes 2B or 4B of storage depending on machine size.
- The classes of integer are:

Unsigned			Signed		
Data type	**Keyword**	**Size**	**Data type**	**Keyword**	**Size**
Short Integer	short int	1B	Signed short integer	signed short int	1B
Integer	int	2B	Signed integer	signed int	2B
Long Integer	long int	4B	Long integer	long int	4B

2. Floating point data type:

- It is used to store decimal numbers.
- The keyword float is used to declare variable of floating point data type.
- Float type takes 4B or 8B of storage depending on machine size.
- It can be expressed in fractional or exponential form.

Data type	Keyword	Size
Floating point	Float	4B
Double	Double	8B
Long double	long double	10B

3. Double:

- It is used to store double precision floating point numbers.
- The keyword double is used to declare variable of floating point data type.
- Double type takes 8B of storage.
- Double precision is related to accuracy of data.

4. Char:

- It is used to store character type of data.
- The keyword char is used to declare variable of character type.
- Char type takes 1B of storage.

5. **Void:**
   - It is a special data type that has no value.
   - It doesn't have size.
   - It is used for returning the result of function that returns nothing.

Variable

A variable is a name given to memory location whose value changes duringexecution.

Types of variables:

1. **Numeric variable:** numeric variable can be used to store either integer values or floating point values.

2. **Character variable:**
   Character variable can include any letter from the alphabet or from the ASCII chart and number from 0-9 that are given within single quotes.

Declaration:

Syntax: datatype variable_list;

Where,

datatype → Any built in data type

variable_list → Identifiers that specifies variable name.

Example: int a;

Rules for declaring/naming variables:

1. Name should only consist of alphabets (both upper and lower case),digits and underscore sign.
2. First characters should be alphabet or underscore.
3. Name should not be a keyword.
4. Since C is a case sensitive the upper case and lower case considered differently for example code ,Code<CODE etc..Are different identifiers.
5. As C defined up to 32 significant characters can be used and will be considered significant by most compilers. If more than 32 are used they will be ignored by the compiler.
6. No white space is used.

Initialization:

Syntax: datatype variable_name = value;Where, datatype → Any built in data type

variable_name → Identifier that specifies variable name.

value →The data which will be stored in variable name.

Example: int a = 5;

Difference between variable and identifiers:

Identifiers	Variables
All identifiers are not variables	All variables are identifiers
Identifiers may not have any memory unless it is a variable	All variables have memory.
Mentioning the type of an identifier it not needed unless it is a variable	Type of the variable must be defined.

Constants:

- Constants or a literal refers to fixed values that do not change during the execution of a program.
- Constants are treated just like regular variables except that their values cannot be modified after their definition.
- C supports several types of constants as shown

Numeric constants:

Numeric constants consist of numeric digits they may or may not have decimal point.

The rules for defining numeric constants are as follows:

1. Numeric constant should have at least one digit.
2. No comma or space is allowed within the numeric constant.
3. Numeric constant can either be positive or negative but default sign is always positive.
4. The value of a constant cannot exceed specified minimum and maximum bounds.

There are two types of numeric constants namely, integer constant floating point constant.

1. Integer constants:

Integer constants are whole numbers without any fractional part or decimal point. It must have at least one digit and may contain either + or – sign. A number with no sign is assumed to be positive.

2.Floating point constant or real constant:

- A floating point constant is a base 10 number that contains either a decimal point or anexponent or both. It may also have either + or – sign preceding it.

Example: 0.05,-0. 09 , 33. 44

- The interpretation of a floating point constant with an exponent is essentially the same asscientific notation, except that the base 10 is replaced by the letter E or e.
- Thus the number 1.2X 10^-3 would be written as 1.2E-3 or 1.2e-3.

2. Character constants:

A character constant contains one single character enclosed within single quotes (‘’)

Example:

‘a’, ’Z’,’\n’

3. String constants:

String constant are sequence of character enclosed within double quotes (‘””).

Example:
"hello","1234","abc","\n"

Declaring constants:

1. To declare a constant precede the normal variable declaration with const keyword and assign it a value.
   Syntax: const data type constant value
   Example: const float Pi=3.142
2. Another way is to use the preprocessor command define.
   Syntax: #define const value
   Example:
   #define PI 3.142

Input and output statements/Formatted I/O statements:

Input function: scanf():

We use the scanf() function for getting the formatted inputs or standard inputs so that the printf () function can provide the program with numerous options of conversion.

Syntax for scanf()

The purpose of the scanf() function is to read the characters that we get from the standard input, convert them according to the string of format specification, and then store the available inputs in the memory slots that the other arguments represent

Inputting Values Using scanf- To enter the input through the input devices like keyboard we make use of scanf statement.

General Syntax:

scanf (format_specifier, &data_a, &data_b,……); // Here, & refers to the address operator

- Where: Format string consists of the access specifiers/format specifiers.
- Format string also called as control string.
- Format string consist of format specifier of particular data type
- Format specifiers starts with % sign followed by conversion code.
- List of addresses of variables consist of the variable name preceded with & symbol(address operator).

Example:int a; float b;
scanf("%d%f",&a,&b);

Rules for scanf

- No escape sequences or additional blank spaces should be specified in the format specifiers.
  Ex: scanf("%d %f",&a,&b); //invalid scanf("%d\n%f",&a,&b); //invalid
- & symbol is must to read the values, if not the entered value will not be stored in the variable specified. Ex: scanf("%d%f",a,b);//invalid.

Output statement: Printf():

We use this function for displaying a single or multiple values in the form of output for the user end at the console.

- printf is an output statement in C used to display the content on the screen.
- print: Print the data stored in the specified memory location or variable.
- Format: The data present in memory location is formatted in to appropriate data type.
- There are various forms of printf statements.

Method 1: printf(" format string");

Format string may be any character. The characters included within the double quoteswill be displayed on the output screen

Example: **printf("Welcome to India");**

Output:

```
Welcome to India
```

Method 2: printf(" format string", variable list);

- Format string also called as control string.
- Format string consist of **format specifier** of particular data type
- Format specifiers starts with **%** sign followed by **conversion code.**
- variable list are the **variable names** separated by **comma.**

Example:

int a=10; float b=20;

printf(" integer =%d, floating=%f",a,b);

output:

```
integer=10, floating=20.00000
```

- Number of format specifiers must be equal to number of variables.
- While specifying the variables name make sure that it matches to the **formatspecifiers** with in the double quotes.

Format Specifiers

- Format specifiers are the character string with **% sign** followed with a character.
- It specifies the type of data that is being processed.
- It is also called ***conversion specifier or conversion code*.**
- There are many format specifiers defined in C.

Symbols	Meaning
%d	Decimal signed integer number
%f	float point number
%c	Character
%o	octal number
%x	hexadecimal integer(Lower case letter x)
%X	hexadecimal integer(Upper case letter X)
%e	floating point value with exponent(Lower case letter e)
%E	floating point value with exponent (Upper case letter E)
%ld	long integer
%s	String
%lf	double

Program to illustrate input and output statements:

```
#include <stdio.h>

int main()
{
 int i;
 float f;
 char c;

 printf("Enter an integer and a float, then Y or N\n> ");
 scanf("%d%f%c", &i, &f, &c);

 printf("You entered:\n");
 printf("i = %d, f = %f, c = %c\n", i, f, c);

}
```

Questions

1. Define computer. Describe the various types of computers based on speed, memory and cost.
2. Discuss the variants of microcomputer that are widely used today.
3. Explain the components of a computer with a neat diagram.
4. Write a short note on the characteristics of a computer.
5. Mention various output devices and explain hardcopy devices. Describe any three input devices
6. Develop an algorithm to find the area and perimeter of a circle. Also define an algorithm.
7. Draw a flowchart and C program which takes as input p,t,r. Compute the simple interest and display the result.
8. Design an algorithm, flowchart and program to compute area of a circle.
9. Explain the structure of C program in detail. Write a sample program to demonstrate the components in the structure of C program.
10. Discuss different types of error occur in program.
11. What is variable? What are the rules to construct variable? Classify the following as valid/invalid Identifiers. i) num2 ii) $num1 iii) +add iv) a_2 v) 199_space vi) _apple vii)#12
12. Write a note on the following operators. i) Relational ii) Logical iii) Conditional
13. Demonstrate formatted output of integer in C with suitable example.
14. Summarize the formatted input and output statements with suitable syntax and example
15. Explain the SDLC life cycle for the efficient design of a program with a neat diagram.
16. What are the basic datatypes available in C.