ACMT Group of Colleges

Polytechnic- 2nd Year/ 3rdSem

COMPUTER SCIENCE

INTRODUCTION TO PROGRAMMING & C

By-Abhishek Tyagi

What is a computer?

A **computer** is an electronic device that manipulates information, or data. It has the ability to **store**, **retrieve**, and **process** data.

A computer is a programmable electronic device that accepts raw data as input and processes it with a set of instructions (a program) to produce the result as output.

Computers are divided into different types based on different criteria. Based on the size, a computer can be divided into five types:

1. Micro Computer
2. Mini Computer
3. Mainframe Computer
4. Super Computer

1. Micro Computer:

It is a single-user computer which has less speed and storage capacity than the other types. It uses a microprocessor as a CPU. The first microcomputer was built with 8-bit microprocessor chips. The common examples of microcomputers include laptops, desktop computers, personal digital assistant (PDA), tablets, and smartphones. Microcomputers are generally designed and developed for general usage like browsing, searching for information, internet, MS Office, social media, etc.

2. Mini Computer:

Mini-computers are also known as "Midrange Computers." They are not designed for a single. They are multi-user computers designed to support multiple users simultaneously. So, they are generally used by small businesses and firms. Individual departments of a company use these computers for specific purposes. For example, the admission department of a University can use a Mini-computer for monitoring the admission process.

3. Mainframe Computer:

It is also a multi-user computer capable of supporting thousands of users simultaneously. They are used by large firms and government organizations to run their business operations as they can store and process large amounts of data. For example, Banks, universities, and insurance companies use mainframe computers to store the data of their customers, students, and policyholders, respectively.

4. Super Computer:

Super-computers are the fastest and most expensive computers among all types of computers. They have huge storage capacities and computing speeds and thus can perform millions of instructions per second. The super-computers are task-specific and thus used for specialized applications such as large-scale numerical problems in scientific and engineering disciplines including applications in electronics, petroleum engineering, weather forecasting, medicine, space research and more. For example, NASA uses supercomputers for launching space satellites and monitoring and controlling them for space exploration.

Classification of Computer

Analog Computer

An analog computer (spelt analogue in British English) is a form of computer that uses *continuous* physical phenomena such as electrical, mechanical, or hydraulic quantities to model the problem being solved.

Digital Computer

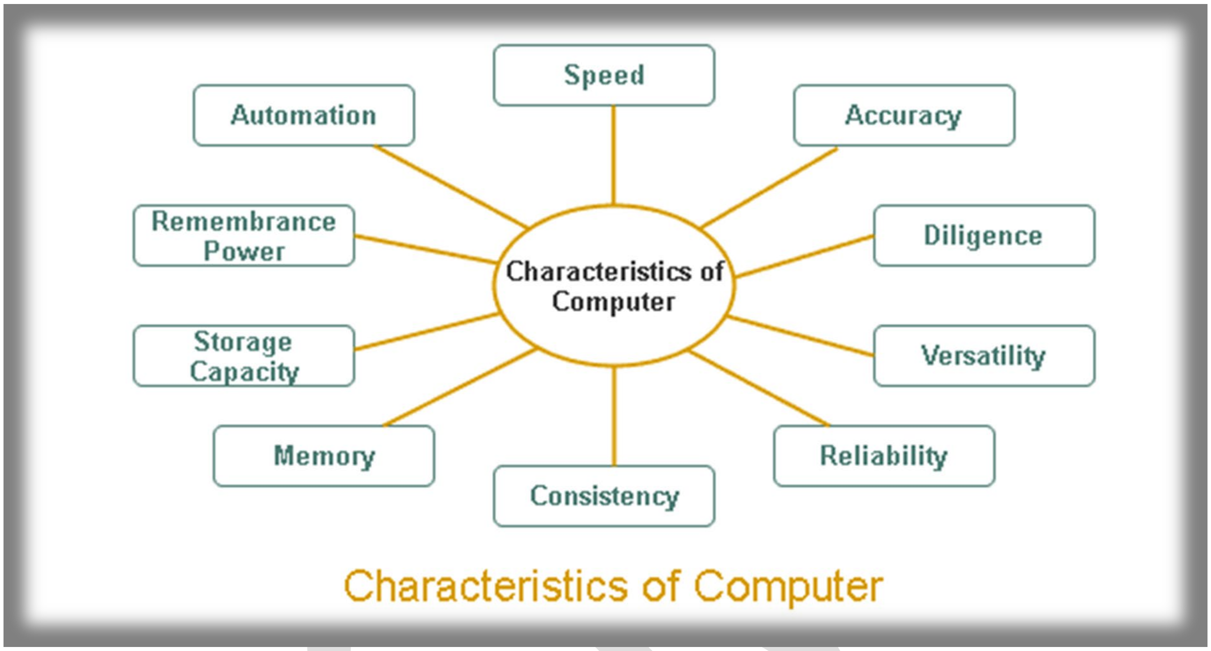
A computer that performs calculations and logical operations with quantities represented as digits, usually in the binary number system

Hybrid Computer (Analog + Digital)

A combination of computers those are capable of inputting and outputting in both digital and analog signals. A hybrid computer system setup offers a cost effective method of performing complex simulations.

What are the Characteristics of Computer?

There are various characteristics of computer system depending on their size, capacity, and specifications. But, the major characteristics of the computer can be classified into **Speed, Accuracy, Diligence, Versatility, Reliability, Consistency, Memory, Storage Capacity, Remembrance Power, and Automation:**



Functions of Computers

A computer performs the following functions −

Receiving Input

Data is fed into computer through various input devices like keyboard, mouse, digital pens, etc. Input can also be fed through devices like CD-ROM, pen drive, scanner, etc.

Processing the information

Operations on the input data are carried out based on the instructions provided in the programs.

Storing the information

After processing, the information gets stored in the primary or secondary storage area.

Producing output

The processed information and other details are communicated to the outside world through output devices like monitor, printer, etc.

Block Diagram Of COmputer

Input

All the data received by the computer goes through the input unit. The input unit comprises different devices. Like a mouse, keyboard, scanner, etc. In other words, each of these devices acts as a mediator between the users and the computer.
The data that is to be processed is put through the input unit. The computer accepts the raw data in binary form. It then processes the data, and produces the desired output.

The 3 major functions of the input unit are-

- Take the data to be processed by the user.
- Convert the given data into machine-readable form.
- And then, transmit the converted data into the main memory of the computer. The sole purpose is to connect the user and the computer. In addition, this creates easy communication between them.

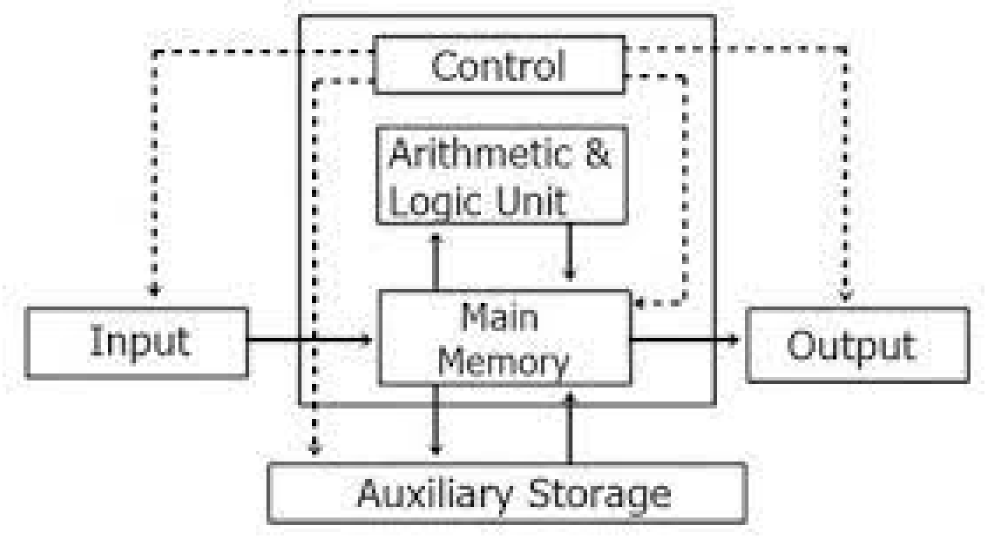
CPU – Central Processing Unit

Central Processing Unit or the CPU, is the brain of the computer. It works the same way a human brain works. As the brain controls all human activities, the CPU too controls all tasks.

Moreover, the CPU conducts all the arithmetical and logical operations in the computer.

Now the CPU comprises of two units, namely – ALU (Arithmetic Logic Unit) and CU (Control Unit). Both of these units work in sync. The CPU processes the data as a whole.

Let us see what particular tasks are assigned to both units.



Block Diagram of Computer

ALU – Arithmetic Logic Unit

The Arithmetic Logic Unit is made of two terms, arithmetic and logic. There are two major functions that this unit performs.

1. Data inserted through the input unit into the primary memory. Performs the basic arithmetical operation on it. Like addition, subtraction, multiplication, and division. It performs all sorts of calculations required on the data. Then sends back data to the storage.
2. The unit is also responsible for performing logical operations like, AND, OR, Equal to, Less than, etc. In addition to this it conducts merging, sorting, and selection of the given data.

CU – Control Unit

The control unit as the name suggests is the controller of all the activities/tasks and operations. All this is performed inside the computer.

The memory unit sends a set of instructions to the control unit. Then the control unit in turn converts those instructions. After that these instructions are converted to control signals.

These control signals help in prioritizing and scheduling the activities. Thus, the control unit coordinates the tasks inside the computer in sync with the input and output units.

Memory Unit

All the data that has to be processed or has been processed is stored in the memory unit. The memory unit acts as a hub of all the data. It transmits it to the required part of the computer whenever necessary.

The memory unit works in sync with the CPU. This helps in faster accessing and processing of the data. Thus, making tasks easier and faster.

There are two types of computer memory-

1. **Primary memory** – This type of memory cannot store a vast amount of data. Therefore, it is only used to store recent data. The data stored in this is temporary. It can get erased once the power is switched off. Therefore, is also called temporary memory or the main memory.

   RAM stands for Random Access Memory. It is an example of primary memory. This memory is directly accessible by the CPU. It is used for reading and writing purposes. For data to be processed, it has to be first transferred to the RAM and then to the CPU.

2. **Secondary memory** – As explained above, the primary memory stores temporary data. Thus it cannot be accessed in the future. For permanent storage purposes, secondary memory is used. It is also called the permanent memory or the auxiliary memory. The hard disk is an example of secondary memory. Even in a power failure data does not get erased easily.

Output

There is nothing to be amazed by what the output unit is used for. All the information sent to the computer once processed is received by the user through the output unit. Devices like printers, monitors, projector, etc. all come under the output unit.

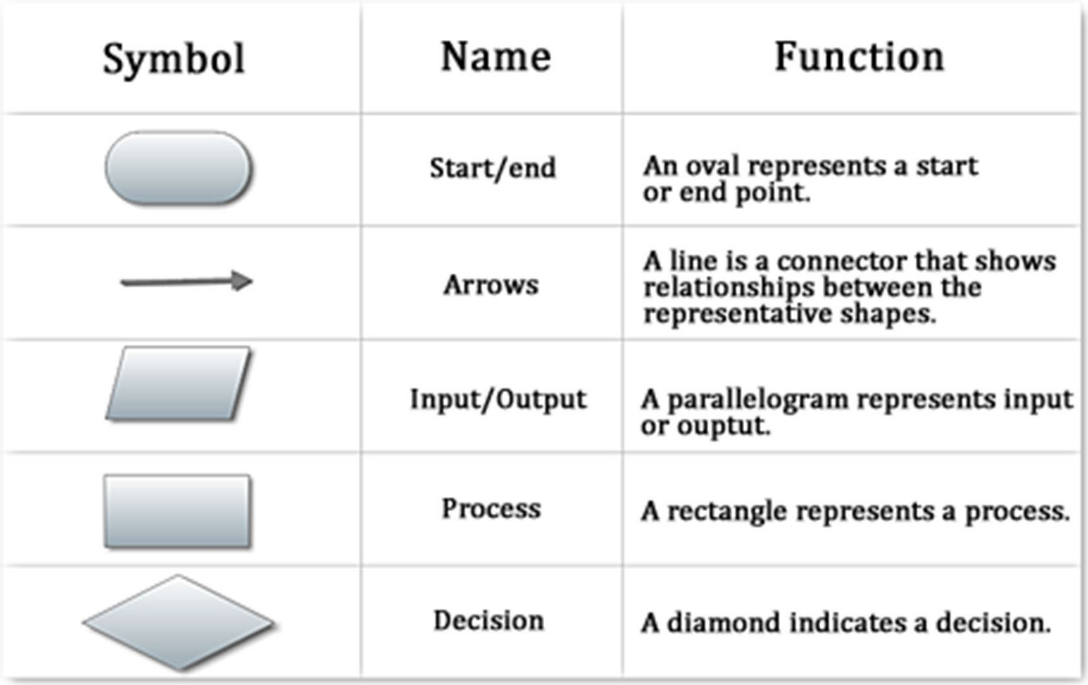
The output unit displays the data either in the form of a soft copy or hard copy. The printer is for the hard copy. The monitor is for the display. The output unit accepts the data in binary form from the computer. It then converts it into a readable form for the user.

Algorithm

The current term of choice for a problem-solving procedure, *algorithm*, is commonly used nowadays for the set of rules a machine (and especially a computer) follows to achieve a particular goal. It does not always apply to computer-mediated activity, however. The term may as accurately be used of the steps followed in making a pizza or solving a Rubik's Cube as for computer-powered data analysis.

What is a Flowchart?

A flowchart is a visual representation of the sequence of steps and decisions needed to perform a process. Each step in the sequence is noted within a diagram shape. Steps are linked by connecting lines and directional arrows. This allows anyone to view the flowchart and logically follow the process from beginning to end.



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 ouptut.
▭	Process	A rectangle represents a process.
◇	Decision	A diamond indicates a decision.

Characteristics of an Algorithm

The following are some key characteristics of an algorithm.

1. Each step of an algorithm must be exact; this means that an algorithm must be precise and unambiguously described. This eliminates any uncertainty. It can also be said to the characteristic of precision, i.e. the steps are precisely stated (defined).
2. Algorithms must terminate; since the ultimate aim of an algorithm is to solve a problem, then it must terminate otherwise there won't be a solution for the problem. This leads to the fact that an algorithm must have a finite number of steps in its execution. The presence of endless (infinite) loops must be avoided.
3. An algorithm must be effective; this means that an algorithm must provide the correct answers at all times.
4. An algorithm must be general; this means that an algorithm must solve every instance of a problem.
5. Uniqueness: results of each step are uniquely defined and only depend on the input and the result of the preceding steps.
6. Finiteness: the algorithm stops after a finite number of instructions are executed.
7. Output: the algorithm produces output.

INPUT DEVICES

1. Keyboard

The most basic input device to enter data on the system or any other device with the help of the keys is a keyboard. They establish a connection with the computer either by wifi or by a USB system. There are keys for everything – numbers, letters, characters, and functions.

The typing keys are A – Z and number keys till 09. The numeric keypad has 17 keys allowing users to perform different calculations. Then there are function keys for special computer features at the top row. The control keys have arrow signs on them allowing users to scroll on the page and select. Then there are special-purpose keys like space, enter, shift, etc.

There are several types of keyboards according to geographic location but the most popular one is QWERTY. The go-to keyboard of the modern-day, even the Latin-based countries use this type of keyboard very frequently. The top six letters of the keyboard are qwerty, giving us the name of the keyboard. Many people even think that this is the only keyboard that exists.

2. Mouse

A hand-supported input device that allows users to move the cursor on the screen is a mouse. It works on a flat surface with a wheel between the left and right buttons. Laptops have a touchpad as does the function of a mouse. The invention of the mouse took place in 1063 by Douglas C. Engelbart.
The earliest version of the mouse had a rollerball underneath but the modern ones are made with optical technology with a light beam. The port of the mouse connection depends on the computer and mouse type. Its advantages are –

- Easy to use
- Affordable
- Moves the cursor faster

3. Joystick

Joystick is a pointing device to move the cursor on the monitor screen but not the same as a mouse. It has a spherical ball stick-type structure. It can move in all directions and is a computer aid design mainly for gaming purposes.

4. Light Pen

Again a pointing device with a pen-like structure allowing users to select on the screen and even draw on it. Photocells and optical systems are inbuilt inside the small tube allowing the flow of signals to the CPU. The tip of the pen moves on the screen with the button pressed, the CPU receives the signals.

5. Track Ball

A common accessory for notebooks and laptops, that works as a replacement for a mouse. With a similar structure, it has a half inserted ball which the fingers move to send signals for cursor movement. It is kind of stationary taking very little space. It comes in a ball, button, and square shape as well.

6. Scanner

A scanner works like a photocopy machine is an input device to make information available on a pc from paper. This is mainly to manipulate the information by converting an image into a digital form and then printing it.

7. Digitizer

Responsible for converting analog information into digital form, a digitizer converts signals into numeric values stored in a computer. The computer can now do anything with it as long as the camera is pointing out the image. A graphic tablet is also a digitizer converting graphics into binary data. It is ideal for drawing and image manipulation applications.

8. Microphone

It is an inbuilt voice input device to take in the sound signals and then convert them into digital form. It is a very common device present in multimedia presentation and music-related applications.

9. Magnetic Ink Card Reader (MICR)

MICR is a device commonly present in banks as they deal with many cheques every day. The bank's code number and cheque number have magnetic ink present in them allowing machines to read the letters. This process is Magnetic Ink Character Recognition that has fewer errors and is very fast.

10. Optical Character Reader (OCR)

It is also a reading device that reads printed text by scanning it character by character. It first converts them into machine-readable code and saves them in the system memory.

11. Bar Code Readers

Again a reading device but to read barcode data like on goods, books, etc. It can be a handheld scanner or a stationary one but they both scan the image by converting it to alphanumeric value on the computer.

12. Optical Mark Reader (OMR)

It is an optical scanner to recognize marks by pen and pencil and is commonly present in educational institutions to check objective exam papers.

OUTPUTDEVICES

1. Speakers

An output device that produces sound after receiving a command from the computer. They support the computers as well as other hardware devices. Now we also have wireless speakers with Bluetooth technology.

2. Projector

This is an optical device that presents visuals on the screen, stationary and moving both. They are present at movie theatres, auditoriums, etc. It connects to the computer and displays the image on it on a larger screen.

3. Plotter

It is a device to make graphics, prints and other vector images to give a real life illustration vibe. It is mandatory to have a graphic card to use the device. The pen like device that comes with it helps in replicating the exact design on the computer.

4. Television

A very common output device present in most of the households is a display output device. It portrays video and audio files on the screen according to the user's needs. Before we had CRT screens but now most of us use plasma displays.

8. Video Card

This device goes inside the sockets of a motherboard of the computer system. It improvises the digital content appearance in other output devices. It is very important now with many people having multiple devices just for personal use.

5. Global Positioning System

A device to help users with directions, GPS uses satellite technology to track geographical location of the users. There is a continuous latitudinal and longitudinal calculation to have an accurate result every time. Most of the vehicles and smart devices have GPS as an inbuilt feature.

6. Headphones

Similar to speakers, this device has a lower sound frequency. They can’t be easily heard in large areas like grounds and parks but are only accessible to a single person using the device. A headset is another name for them.

7.Monitore

The visual display units are the most important output device responsible for showing the visual made of pixels to the user. The pixels decide the image sharpness.

8.Printers

a. Impact Printers

They have a ribbon on the front which prints characters on the paper by pressing it.
Characteristics –

- Consumable cost is less
- Noisy in nature
- Good for bulk printing
- Physical contact between paper and machine

b. Character Printers

These printer print only one character at a time and are of two types –

i. Dot Matrix Printer

They have a very economical price and are easy to use. Each character print has a dots pattern of the size (5*7, 7*9, 9*7, or 9*9) creating a complete symbol.
Its advantages are –

- Inexpensive
- Accessible
- Multilingual Printing

Its disadvantages are –

- Slow Speed
- Quality is poor

ii. Daisy Wheel

With head lying on the wheel, the pins next to the characters form a petal shape thus giving it its name. They print nice quality characters and are mostly present at word-processing offices that need a few letters only.

Its advantages are –

- Reliable than the dot matrix printer
- Quality Characters
- Exchangeable Fonts

Its disadvantages are –

- Slow in speed

- Noisy Nature
- A bit expensive
-

c. Line Printers

They print one line at a time. They are of two types –

Drum Printer

They have a drum-like shape with track division equal to paper size, like 100 width paper will have 100 tracks. Each track has a character set embossed either 48 character set, 64 or 96 characters set. Each drum prints one line and is as fast as 300 to 2000 lines per minute.
Its advantages are –

- High speed

Its disadvantages are

- Very Costly
- Only one set of font available in one printer
- Don't make noise
- High-Quality printing
- Many fonts and character size available

MEMORY

A memory is just like a human brain. It is used to store data and instructions. Computer memory is the storage space in the computer, where data is to be processed and instructions required for processing are stored. The memory is divided into large number of small parts called cells. Each location or cell has a unique address, which varies from zero to memory size minus one. For example, if the computer has 64k words, then this memory unit has 64 * 1024 = 65536 memory locations. The address of these locations varies from 0 to 65535.

Memory is primarily of three types −

- Cache Memory
- Primary Memory/Main Memory
- Secondary Memory

Cache Memory

Cache memory is a very high speed semiconductor memory which can speed up the CPU. It acts as a buffer between the CPU and the main memory. It is used to hold those parts of data and program which are most frequently used by the CPU. The parts of data and programs are transferred from the disk to cache memory by the operating system, from where the CPU can access them.

Advantages

The advantages of cache memory are as follows −

- Cache memory is faster than main memory.
- It consumes less access time as compared to main memory.
- It stores the program that can be executed within a short period of time.
- It stores data for temporary use.

Disadvantages

The disadvantages of cache memory are as follows −

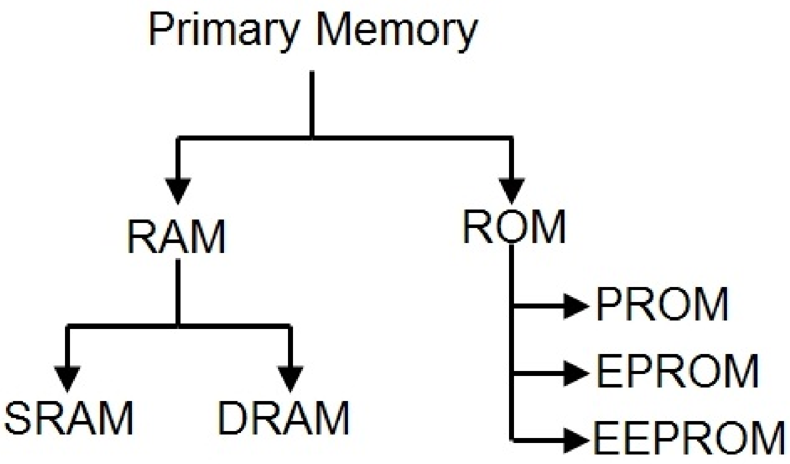
- Cache memory has limited capacity.
- It is very expensive.

Primary Memory (Main Memory)

Primary memory holds only those data and instructions on which the computer is currently working. It has a limited capacity and data is lost when power is switched off. It is generally made up of semiconductor device. These memories are not as fast as registers. The data and instruction required to be processed resides in the main memory. It is divided into two subcategories RAM and ROM.

Characteristics of Main Memory

- These are semiconductor memories.
- It is known as the main memory.
- Usually volatile memory.
- Data is lost in case power is switched off.
- It is the working memory of the computer.
- Faster than secondary memories.
- A computer cannot run without the primary memory.



Sr. No.	Key	RAM	ROM
1	Definition	RAM stands for Random Access Memory.	ROM stands for Read Only Memory.
2	Data Retention	RAM data is volatile. Data is present till power supply is present.	ROM data is permanent. Data remains even after power supply is not present.
3	Data Access	RAM data can be read, erased or modified.	ROM data is readonly.
4	Usage	RAM is used to store data that CPU needs for current instruction processing.	ROM is used to store data that in needed to bootstrap the computer.
5	Speed	RAM speed is quite high.	ROM speed is slower than RAM.
6	CPU Access	CPU can access data stored on RAM.	Data to be copied from ROM to RAM so that CPU can access its data.

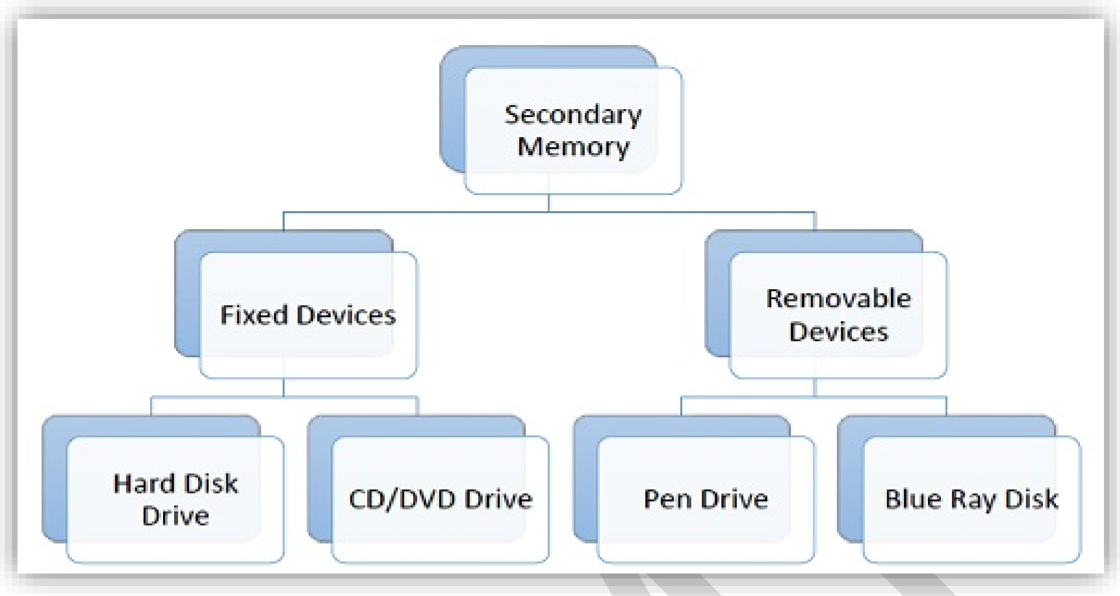
Sr. No.	Key	RAM	ROM
7	Capacity	RAM memory is large and high capacity.	ROM is generally small and of low capacity.
8	Usage	RAM is used as CPU Cache, Primary Memory.	ROM is used as firmware by microcontrollers.
9	Cost	RAM is costly.	ROM is cheap.

Secondary Memory

This type of memory is also known as external memory or non-volatile. It is slower than the main memory. These are used for storing data/information permanently. CPU directly does not access these memories, instead they are accessed via input-output routines. The contents of secondary memories are first transferred to the main memory, and then the CPU can access it. For example, disk, CD-ROM, DVD, etc.

Characteristics of Secondary Memory

- These are magnetic and optical memories.
- It is known as the backup memory.
- It is a non-volatile memory.
- Data is permanently stored even if power is switched off.
- It is used for storage of data in a computer.
- Computer may run without the secondary memory.
- Slower than primary memories.



SECONDARY MEMORY EXAMPLES

1. Floppy Disk 2. Hard Disk Drive 3. Magnetic Tape b Optical discs 1. CD 2. DVD 3. Blu-ray Disc c Solid State 1. Pen/Flash Drive2. Memory Card

Number systems

Number systems are the technique to represent numbers in the computer system architecture, every value that you are saving or getting into/from computer memory has a defined number system.

Computer architecture supports following number systems.

- **Binary number system**
- **Octal number system**
- **Decimal number system**
- **Hexadecimal (hex) number system**

1) Binary Number System

A Binary number system has only two digits that are **0 and 1**. Every number (value) represents with 0 and 1 in this number system. The base of binary number system is 2, because it has only two digits.

2) 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.

3) Decimal number system

Decimal number system has only ten (10) digits from **0 to 9**. Every number (value) represents with 0,1,2,3,4,5,6, 7,8 and 9 in this number system. The base of decimal number system is 10, because it has only 10 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 has 16 alphanumeric values. Here **A is 10**, **B is 11**, **C is 12**, **D is 13**, **E is 14**and **F is 15**.

HISTORY OF C PROGRAMMING-

In 1972, a great computer scientist Dennis Ritchie created a new programming language called ‘C’ at the Bell Laboratories. It was created from ‘ALGOL’, ‘BCPL’ and ‘B’ programming languages. ‘C’ programming language contains all the features of these languages and many more additional concepts that make it unique from other languages.

‘C’ is a powerful programming language which is strongly associated with the UNIX operating system. Even most of the UNIX operating system is coded in ‘C’. Initially ‘C’ programming was limited to the UNIX operating system, but as it started spreading around the world, it became commercial, and many compilers were released for cross-platform systems. Today ‘C’ runs under a variety of operating systems and hardware platforms. As it started evolving many different versions of the language were released. At times it became difficult for the developers to keep up with the latest version as the systems were running under the older versions. To assure that ‘C’ language will remain standard, American National Standards Institute (ANSI) defined a commercial standard for ‘C’ language in 1989. Later, it was approved by the International Standards Organization (ISO) in 1990. ‘C’ programming language is also called as ‘ANSI C’.

Characteristics of C

- Procedural Language.
- Fast and Efficient.
- Modularity.
- Statically Type.
- General-Purpose Language.
- Rich set of built-in Operators.
- Libraries with rich Functions.
- Middle-Level Language.
- Large number of data types.
- Large number of character set.
- Portable (Platform Independent).
- Flexible.
- Structured Programming.

C Character Set

As every language contains a set of characters used to construct words, statements, etc., C language also has a set of characters which include **alphabets, digits**, and **special symbols**. C language supports a total of 256 characters. Every C program contains statements. These statements are constructed using words and these words are constructed using characters from C character set. C language character set contains the following set of characters...

1. Alphabets
2. Digits
3. Special Symbols

Alphabets

C language supports all the alphabets from the English language. Lower and upper case letters together support 52 alphabets.

lower case letters - **a to z**

UPPER CASE LETTERS - **A to Z**

Digits

C language supports 10 digits which are used to construct numerical values in C language.

Digits - **0, 1, 2, 3, 4, 5, 6, 7, 8, 9**

Special Symbols

C language supports a rich set of special symbols that include symbols to perform mathematical operations, to check conditions, white spaces, backspaces, and other special symbols.

Special Symbols - **~ @ # $ % ^ & * ( ) _ - + = { } [ ] ; : ' " / ? . > , < \ | tab newline space NULL bell backspace verticaltab etc.,**

Data Types

Basic Data Type	int, char, float, double
Derived Data Type	array, pointer, structure, union

Keywords in C

A keyword is a **reserved word**. You cannot use it as a variable name, constant name, etc. There are only 32 reserved words (keywords) in the C language.

A list of 32 keywords in the c language is given below:

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

HEADER FILE

In C language, header files contain the set of predefined standard library functions. The “#include” preprocessing directive is used to include the header files with “.h” extension in the program.

Here is the table that displays some of the header files in C language,

Sr.No.	Header Files & Description
1	**stdio.h** Input/Output functions
2	**conio.h** Console Input/Output functions
3	**stdlib.h** General utility functions
4	**math.h** Mathematics functions
5	**string.h** String functions

OPERATORS

An operator is a symbol that tells the compiler to perform specific mathematical or logical functions. C language is rich in built-in operators and provides the following types of operators −

- Arithmetic Operators
- Relational Operators
- Logical Operators
- Bitwise Operators
- Assignment Operators
- Misc Operators

We will, in this chapter, look into the way each operator works.

Arithmetic Operators

The following table shows all the arithmetic operators supported by the C language. Assume variable **A** holds 10 and variable **B** holds 20 then −

Operator	Description	Example
+	Adds two operands.	A + B = 30
−	Subtracts second operand from the first.	A − B = -10
*	Multiplies both operands.	A * B = 200
/	Divides numerator by de-numerator.	B / A = 2
%	Modulus Operator and remainder of after an integer division.	B % A = 0
++	Increment operator increases the integer value by one.	A++ = 11
--	Decrement operator decreases the integer value by one.	A-- = 9

Relational Operators

The following table shows all the relational operators supported by C. Assume variable **A** holds 10 and variable **B** holds 20 then –

Operator	Description	Example
==	Checks if the values of two operands are equal or not. If yes, then the condition becomes true.	(A == B) is not true.
!=	Checks if the values of two operands are equal or not. If the values are not equal, then the condition becomes true.	(A != B) is true.
>	Checks if the value of left operand is greater than the value of right operand. If yes, then the 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 the 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 the 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 the condition becomes true.	(A <= B) is true.

Logical Operators

Following table shows all the logical operators supported by C language. Assume variable **A** holds 1 and variable **B** holds 0, then −

Operator	Description	Example
&&	Called Logical AND operator. If both the operands are non-zero, then the condition becomes true.	(A && B) is false.
||	Called Logical OR Operator. If any of the two operands is non-zero, then the condition becomes true.	(A || B) is true.
!	Called Logical NOT Operator. It is used to reverse the logical state of its operand. If a condition is true, then Logical NOT operator will make it false.	!(A && B) is true.

Bitwise Operators

Bitwise operator works on bits and perform bit-by-bit operation. The truth tables for &, |, and ^ is as follows −

p	q	p & q	p | q	p ^ q
0	0	0	0	0
0	1	0	1	1
1	1	1	1	0
1	0	0	1	1

Assume A = 60 and B = 13 in binary format, they will be as follows −

```
A = 0011 1100

B = 0000 1101

-----------------

A&B = 0000 1100

A|B = 0011 1101
```

A^B = 0011 0001

~A = 1100 0011

The following table lists the bitwise operators supported by C. Assume variable 'A' holds 60 and variable 'B' holds 13, then −

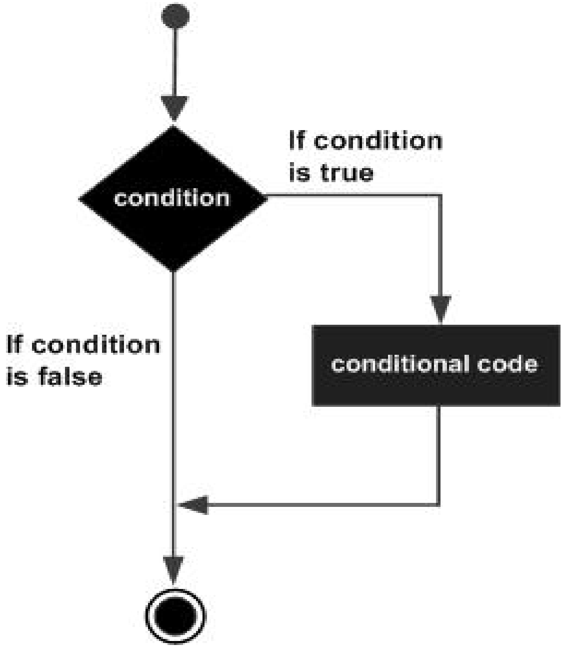
Operator	Description	Example
&	Binary AND Operator copies a bit to the result if it exists in both operands.	(A & B) = 12, i.e., 0000 1100
|	Binary OR Operator copies a bit if it exists in either operand.	(A | B) = 61, i.e., 0011 1101
^	Binary XOR Operator copies the bit if it is set in one operand but not both.	(A ^ B) = 49, i.e., 0011 0001
~	Binary One's Complement Operator is unary and has the effect of 'flipping' bits.	(~A) = ~(60), i.e,. -0111101
<<	Binary Left Shift Operator. The left operands value is moved left by the number of bits specified by the right operand.	A << 2 = 240 i.e., 1111 0000
>>	Binary Right Shift Operator. The left operands value is moved right by the number of bits specified by the right operand.	A >> 2 = 15 i.e., 0000 1111

IF-STATEMENT-

The statements inside the body of “if” only execute if the given condition returns true. If the condition returns false then the statements inside “if” are skipped.

```
if (condition)
{

     //These statements will only execute if the
condition is true
}
```

Flow Diagram of if statement



Example of if statement

```
#include <stdio.h>
int main()
{
    int x = 20;
    int y = 22;
    if (x<y)
    {
        printf("Variable x is less than y");
    }
    return 0;
}
```

Output:

```
Variable x is less than y
```

Explanation: The condition (x<y) specified in the “if” returns true for the value of x and y, so the statement inside the body of if is executed.

FOR LOOP -

A **for** loop is a repetition control structure that allows you to efficiently write a loop that needs to execute a specific number of times.

Syntax

The syntax of a **for** loop in C programming language is −

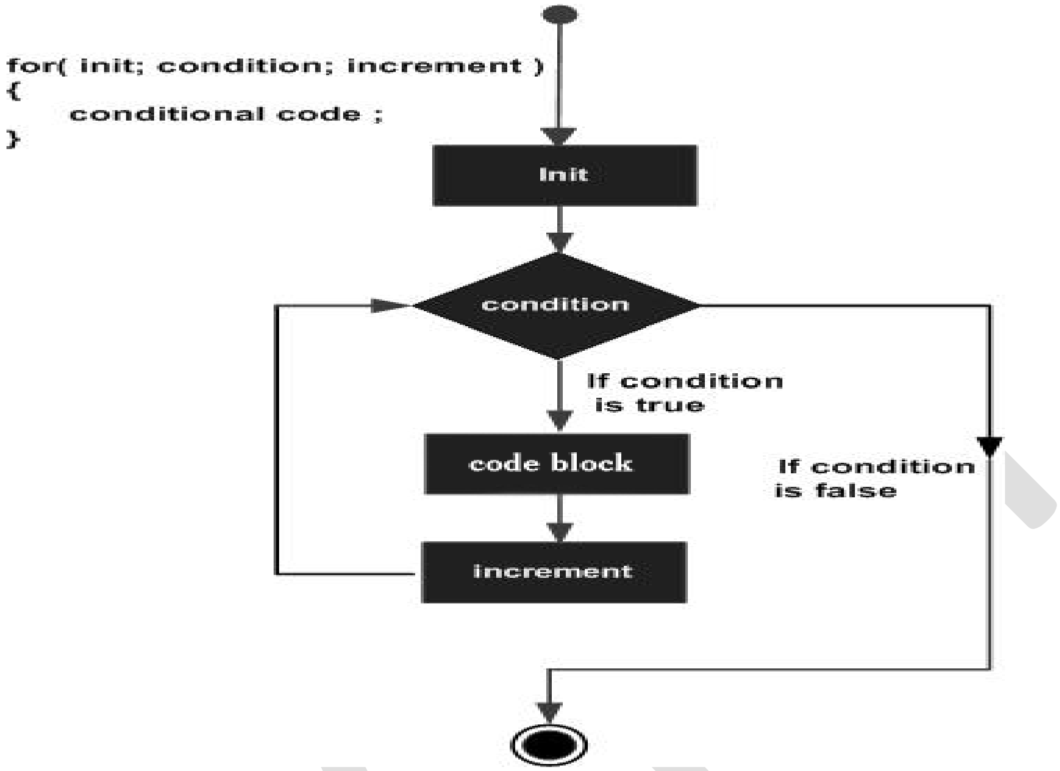
```
for ( init; condition; increment )
{
   statement(s);
}
```

Here is the flow of control in a 'for' loop −

- The **init** step is executed first, and only once. This step allows you to declare and initialize any loop control variables. You are not required to put a statement here, as long as a semicolon appears.
- Next, the **condition** is evaluated. If it is true, the body of the loop is executed. If it is false, the body of the loop does not execute and the flow of control jumps to the next statement just after the 'for' loop.
- After the body of the 'for' loop executes, the flow of control jumps back up to the **increment**statement. This statement allows you to update any loop control variables. This statement can be left blank, as long as a semicolon appears after the condition.

- The condition is now evaluated again. If it is true, the loop executes and the process repeats itself (body of loop, then increment step, and then again condition). After the condition becomes false, the 'for' loop terminates.

Flow Diagram



```
for( init; condition; increment )
{
    conditional code ;
}
```

Example

```
#include <stdio.h>

int main () {

   int a;

   /* for loop execution */
   for( a = 10; a < 20; a = a + 1 ){
      printf("value of a: %d\n", a);
   }

   return 0;
}
```

When the above code is compiled and executed, it produces the following result −

```
value of a: 10
value of a: 11
value of a: 12
value of a: 13
value of a: 14
value of a: 15
value of a: 16
value of a: 17
value of a: 18
value of a: 19
```

Program1.Print Table using for loop

```
#include <stdio.h>
void main()
{
    int num,t, i; // declare a variable
    printf (" Enter a number to generate the table in C: ");
    scanf (" %d", &num); // take a positive number from the user

    printf ("\n Table of %d", num);
    // use for loop to iterate the number from 1 to 10
    for ( i = 1; i <= 10; i++)
    {
        T=n*I,
        printf ("\n %d ", t);
    }
    return 0;
}
```

Output

```
Enter a number to generate the table in C: 7
 Table of 7
 7 * 1 = 7
 7 * 2 = 14
 7 * 3 = 21
 7 * 4 = 28
 7 * 5 = 35
 7 * 6 = 42
 7 * 7 = 49
```

```
7 * 8 = 56
7 * 9 = 63
7 * 10 = 70
```

In the above code, for loop executes from 1 to 10, the given number is multiplied by 1, and the next iteration is multiplied by 2, and the process continues until it multiplies the number to 10. Finally, it prints the table on the screen.

Program2.Factorial Program using loop

Let's see the factorial Program using loop.

```
#include<stdio.h>
int main()
{
 int i,fact=1,number;
 printf("Enter a number: ");
  scanf("%d",&number);
    for(i=1;i<=number;i++){
      fact=fact*i;
  }
  printf("Factorial of %d is: %d",number,fact);
return 0;
}
```

Output:

```
Enter a number: 5
Factorial of 5 is: 120
```

Program3.Fibonacci Series

```
#include<stdio.h>
int main()
{
 int n1=0,n2=1,n3,i,number;
printf("Enter the number of elements:");
scanf("%d",&number);
printf("\n%d %d",n1,n2);//printing 0 and 1
for(i=2;i<number;++i)//loop starts from 2 because 0 and 1 are already printed
{
 n3=n1+n2;
 printf(" %d",n3);
 n1=n2;
 n2=n3;
}
 return 0;
}

```

Program4. Swap two number with third number

```
#include<stdio.h>
 int main()
{
int a=10, b=20;
printf("Before swap a=%d b=%d",a,b);
a=a+b;//a=30 (10+20)
b=a-b;//b=10 (30-20)
a=a-b;//a=20 (30-10)
printf("\nAfter swap a=%d b=%d",a,b);
return 0;
}
```

Output:

```
Before swap a=10 b=20
After swap a=20 b=10
```

Program5.Print right triangle star

```
#include<stdio.h>
#include<conio.h>
```

```
int main()
{
int i, j, n;
printf("Enter number of rows: ");
scanf("%d",&n);
for(i = 1; i <= n; i++)
{
for(j = 1; j <= i; j++)
{
printf("* ");
}
//Ending line after each row
printf("\n");
}
return 0;
}
```

Output-

```
Enter number of rows:5


*

**

***

****

*****
```

SWITCH CASE-

A switch statement allows a variable to be tested for equality against a list of values. Each value is called a case, and the variable being switched on is checked for each switch case.

Syntax

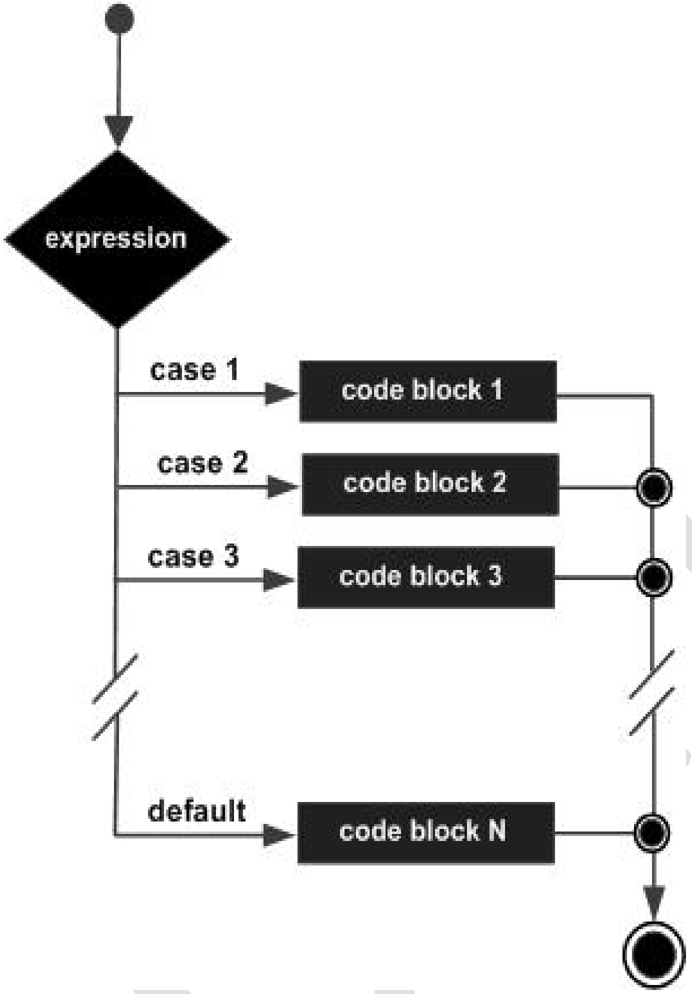
The syntax for a switch statement in C programming language is as follows −

```
switch(expression) {

  case constant-expression  :
    statement(s);
    break; /* optional */

  case constant-expression  :
    statement(s);
    break; /* optional */

  /* you can have any number of case statements */
  default : /* Optional */
  statement(s);
}
```

The following rules apply to a switch statement −

- The expression used in a switch statement must have an integral or enumerated type, or be of a class type in which the class has a single conversion function to an integral or enumerated type.
- You can have any number of case statements within a switch. Each case is followed by the value to be compared to and a colon.
- The constant-expression for a case must be the same data type as the variable in the switch, and it must be a constant or a literal.
- When the variable being switched on is equal to a case, the statements following that case will execute until a break statement is reached.
- When a break statement is reached, the switch terminates, and the flow of control jumps to the next line following the switch statement.
- Not every case needs to contain a break. If no break appears, the flow of control will *fall through* to subsequent cases until a break is reached.
- A switch statement can have an optional default case, which must appear at the end of the switch. The default case can be used for performing a task when none of the cases is true. No break is needed in the default case.

Flow Diagram



Example6.

```
#include <stdio.h>

int main () {

   /* local variable definition */
   char grade = 'B';
```

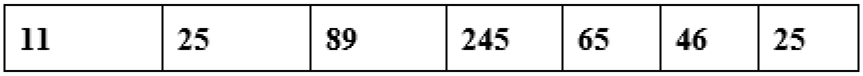
```
  switch(grade) {
     case 'A' :
        printf("Excellent!\n" );
        break;
     case 'B' :
     case 'C' :
        printf("Well done\n" );
        break;
     case 'D' :
        printf("You passed\n" );
        break;
     case 'F' :
        printf("Better try again\n" );
        break;
     default :
        printf("Invalid grade\n" );
  }

  printf("Your grade is  %c\n", grade );

  return 0;
}
```

When the above code is compiled and executed, it produces the following result −

```
Well done
Your grade is B
```

ARRAY-

The array is a type of data structure that is used to store homogeneous data in contiguous memory locations. Following are arrays in C programming.



11	25	89	245	65	46	25

0 1 2 3 4 5 6 ← Index

Here index refers to the location of an element in the array. Let us imagine if A[L] is the name of the array, where “A” is the variable name, and “L” is the length of the array, i.e. the number of elements present in the array.

Then A[i] represents the element at that “i+1”th position in the array, .for example:

A[6]= 72 means element at 6+1 th location of the array.

Declaration of Array-

In C, the array must be declared properly before using it with its name and length. There are three syntaxes in which we can declare arrays in a c program

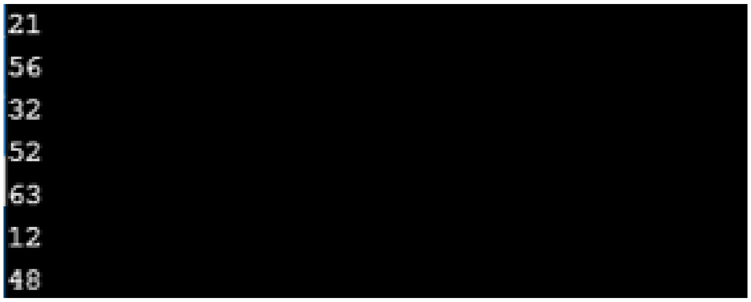
Syntax 1

int A[7] = {21,56,32,52,63,12,48} – Declaring the length and elements of array

Program7:

```
#include<stdio.h>
int main{
int a[7] = {21,56,32,52,63,12,48};
int i;
for(i=0;i<7;i++){
printf(“%d\n”,a[i]);
}
return 0;
}
```

Output:



```
21
56
32
52
63
12
48
```

What is cybercrime?

Cybercrime is any criminal activity that involves a computer, networked device or a network.

cybercrime, also called **computer crime**, the use of a computer as an instrument to further illegal ends, such as committing fraud, trafficking in child pornography and intellectual property, stealing identities, or violating privacy. Cybercrime, especially through the Internet, has grown in importance as the computer has become central to commerce, entertainment, and government.

Types of cyber crime

1. Physical threats

2. Unauthorized person

3. Accidental error

4. Malicious attack

CIA in Cyber Security-

Confidentiality

Keeping data secure

At its core, the tenet of confidentiality is about keeping what needs to be private, private. Government regulation, industry compliance requirements, expectations from your business partners and your company's own business priorities all play a role in defining what data needs to be kept confidential.
In practice, confidentiality is about controlling access to data so that only authorized users can access or modify it. No matter what industry a business is in, it's that business's responsibility to keep their data and their clients'/customers' data out of the

hands of those who would misuse it. This is perhaps then the most obvious of the three CIA components.

Confidentiality can be violated both intentionally and unintentionally, through direct attacks meant to gain access through vulnerable parts of a network or through carelessness and human error. Having strong controls and good training for employees goes a long way in maintaining a business's confidentiality.

Integrity

Keeping data clean

Integrity focuses on keeping data clean and untainted, both when it's uploaded and when it's stored. This means making sure only those who are allowed to modify it, modify it.

While data being leaked is a problem, having data be maliciously or accidentally altered can also create a world of problems and weeks of headaches for businesses. When this happens, trust flies out the window. Businesses, their partners and their customers need to be able to rely on accurate, reliable, up-to-date information at all times. If this cannot be the case, there's a problem.

This requirement isn't just applicable to data that must be kept confidential. Content on a company's website needs to be accurate, too. Pricing, descriptions and even store hours need to all be accurate. This sort of publicly visible data must have its integrity protected as well.

Availability

Keeping data accessible

Availability essentially means that when an authorized user needs to access data or information, they can. It can sometimes be confused with or even seem to contradict confidentiality.

While confidentiality is about making sure that only the people who need to access the data can get to it, availability is about making sure that it's easy to access that data should an authorized person need to. This can include making sure networks and applications are running as they should, that security protocols are not hindering productivity or that a resource is on-hand for when an issue arises and needs fixing.

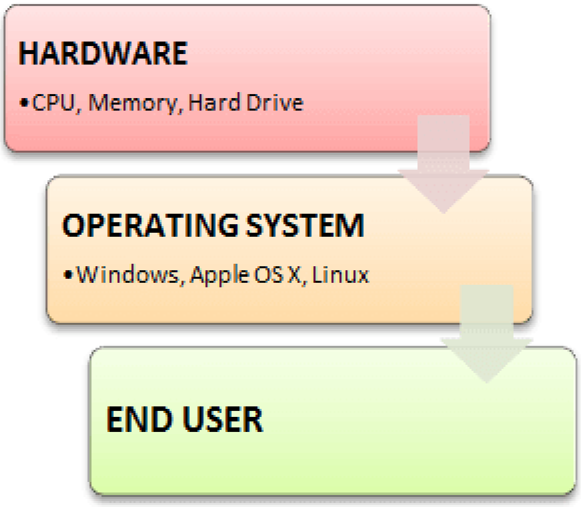
When availability comes under attack or gets left by the wayside, business can come to a halt. Whether it's a block on payroll or email or confidential data required to operate a business, if employees can't get to what they need to work, well, they can't work.

Finding the balance between accessing data and making sure that your business can still operate is a key part of the CIA triad.

Operating System-

An Operating System (OS) is an interface between a computer user and computer hardware. An operating system is a software which performs all the basic tasks like file management, memory management, process management, handling input and output, and controlling peripheral devices such as disk drives and printers.

An operating system is a program that acts as an interface between the user and the computer hardware and controls the execution of all kinds of programs.



Following are some of important functions of an operating System.

- Memory Management
- Processor Management
- Device Management
- File Management
- Security
- Control over system performance
- Job accounting
- Error detecting aids
- Coordination between other software and users

Memory Management

Memory management refers to management of Primary Memory or Main Memory. Main memory is a large array of words or bytes where each word or byte has its own address.

Main memory provides a fast storage that can be accessed directly by the CPU. For a program to be executed, it must in the main memory. An Operating System does the following activities for memory management −

- Keeps tracks of primary memory, i.e., what part of it are in use by whom, what part are not in use.
- In multiprogramming, the OS decides which process will get memory when and how much.
- Allocates the memory when a process requests it to do so.
- De-allocates the memory when a process no longer needs it or has been terminated.

Processor Management

In multiprogramming environment, the OS decides which process gets the processor when and for how much time. This function is called **process scheduling**. An Operating System does the following activities for processor management −

- Keeps tracks of processor and status of process. The program responsible for this task is known as **traffic controller**.
- Allocates the processor (CPU) to a process.
- De-allocates processor when a process is no longer required.

Device Management

An Operating System manages device communication via their respective drivers. It does the following activities for device management −

- Keeps tracks of all devices. Program responsible for this task is known as the **I/O controller**.
- Decides which process gets the device when and for how much time.
- Allocates the device in the efficient way.
- De-allocates devices.

File Management

A file system is normally organized into directories for easy navigation and usage. These directories may contain files and other directions.

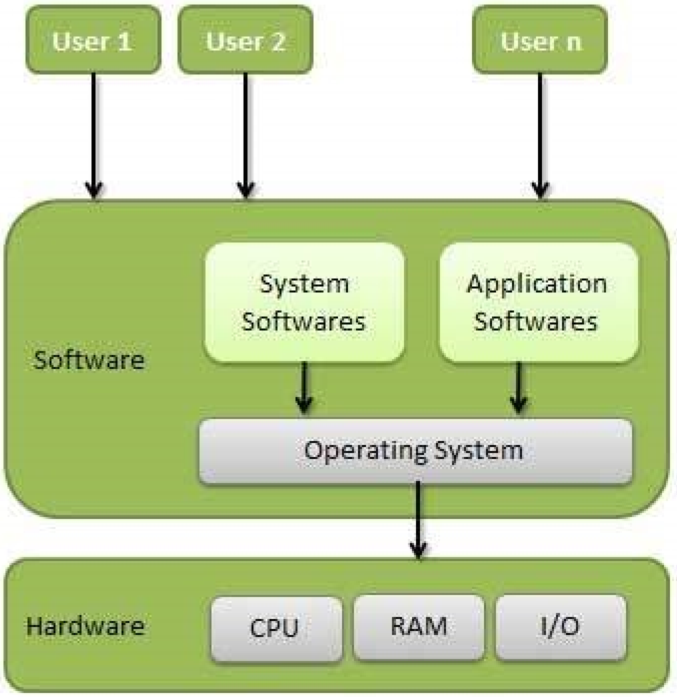
An Operating System does the following activities for file management −

- Keeps track of information, location, uses, status etc. The collective facilities are often known as **file system**.
- Decides who gets the resources.
- Allocates the resources.
- De-allocates the resources.

Other Important Activities

Following are some of the important activities that an Operating System performs −

- **Security** − By means of password and similar other techniques, it prevents unauthorized access to programs and data.
- **Control over system performance** − Recording delays between request for a service and response from the system.
- **Job accounting** − Keeping track of time and resources used by various jobs and users.
- **Error detecting aids** − Production of dumps, traces, error messages, and other debugging and error detecting aids.
- **Coordination between other softwares and users** − Coordination and assignment of compilers, interpreters, assemblers and other software to the various users of the computer systems.



Types of Operating System (OS)

Following are the popular types of OS (Operating System):

- Batch Operating System
- Multitasking/Time Sharing OS
- Multiprocessing OS
- Real Time OS
- Distributed OS
- Network OS
- Mobile OS

Batch Operating System

Some computer processes are very lengthy and time-consuming. To speed the same process, a job with a similar type of needs are batched together and run as a group.

The user of a batch operating system never directly interacts with the computer. In this type of OS, every user prepares his or her job on an offline device like a punch card and submit it to the computer operator.

Multi-Tasking/Time-sharing Operating systems

Time-sharing operating system enables people located at a different terminal(shell) to use a single computer system at the same time. The processor time (CPU) which is shared among multiple users is termed as time sharing.

Real time OS

A real time operating system time interval to process and respond to inputs is very small. Examples: Military Software Systems, Space Software Systems are the Real time OS example.

Distributed Operating System

Distributed systems use many processors located in different machines to provide very fast computation to its users.

Network Operating System

Network Operating System runs on a server. It provides the capability to serve to manage data, user, groups, security, application, and other networking functions.

Mobile OS

Mobile operating systems are those OS which is especially that are designed to power smartphones, tablets, and wearables devices.

Advantage of using Operating System

- Allows you to hide details of hardware by creating an abstraction
- Easy to use with a GUI
- Offers an environment in which a user may execute programs/applications
- The operating system must make sure that the computer system convenient to use
- Operating System acts as an intermediary among applications and the hardware components
- It provides the computer system resources with easy to use format
- Acts as an intermediator between all hardware's and software's of the system

Disadvantages of using Operating System

- If any issue occurs in OS, you may lose all the contents which have been stored in your system
- Operating system's software is quite expensive for small size organization which adds burden on them. Example Windows
- It is never entirely secure as a threat can occur at any time