

**VIJAYABHERI**



**MALAPPURAM DISTRICT PANCHAYATH  
EDUCATIONAL PROJECT**

# **COMPUTER SCIENCE**

**Focus Plus One - 2021**

STUDY MATERIAL FOR PLUS ONE EXAMINATION 2021

**ACADEMIC SUPPORT: ACT MALAPPURAM**

# ആമുഖം

മലപ്പുറം ജില്ലാ പഞ്ചായത്ത് വിജയഭേരി വിദ്യാഭ്യാസ പദ്ധതിയുടെ ഭാഗമായി കഴിഞ്ഞ വർഷം പ്ലസ് ടു / വി.എച്ച്.എസ്.ഇ. രണ്ടാം വർഷ വിദ്യാർത്ഥികൾക്കായി ഫോക്കസ് 21 എന്ന പേരിൽ തയ്യാറാക്കിയ സ്റ്റഡി മെറ്റീരിയലുകൾക്ക് അധ്യാപകരിൽ നിന്നും വിദ്യാർത്ഥികളിൽ നിന്നും വളരെ നല്ല പ്രതികരണമാണ് ലഭിച്ചത്. കോവിഡ് മഹാമാരിമൂലം സാധാരണ ക്ലാസുകൾ ലഭിക്കാത്ത വിദ്യാർത്ഥികൾക്ക് പ്രസ്തുത മെറ്റീരിയൽ ഏറെ സഹായകരമായെന്ന് അവർ സാക്ഷ്യപ്പെടുത്തുന്നു.

ഒന്നാം വർഷ വിദ്യാർത്ഥികൾക്കുള്ള പരീക്ഷ സെപ്റ്റംബർ ആദ്യവാരം നടക്കുകയാണ്. ഫോക്കസ് പാഠഭാഗങ്ങൾക്കായി വിജയഭേരി ഫോക്കസ് +1 എന്ന പേരിൽ കഴിഞ്ഞ വർഷത്തേതു പോലെ ഈ വർഷവും വിവിധ വിഷയങ്ങൾക്ക് പ്രത്യേക സ്റ്റഡി മെറ്റീരിയൽ മലപ്പുറം ജില്ലാ പഞ്ചായത്ത് വിജയഭേരി വിദ്യാഭ്യാസ പദ്ധതിയുടെ ഭാഗമായി പുറത്തിറക്കുകയാണ്. മലപ്പുറം ഡയറ്റാണ് പ്രസ്തുത മെറ്റീരിയലിനുള്ള അക്കാദമിക പിന്തുണ നൽകിയിട്ടുള്ളത്. വിവിധ വിഷയങ്ങളുടെ ജില്ലാ തല അധ്യാപകരുടെ അസോസിയേഷനാണ് ഈ പ്രവർത്തനങ്ങൾക്ക് ഞങ്ങളുടെ കൂടെ നിന്നത്. എല്ലാവരെയും ഈ അവസരത്തിൽ നന്ദിയോടെ സ്മരിക്കുന്നു.

ഈ ഉദ്യമം അധ്യാപകർക്കും വിദ്യാർത്ഥികൾക്കും ഏറെ ഉപകാരപ്പെടുമെന്ന് പ്രതീക്ഷിക്കുന്നു. എല്ലാ വിദ്യാർത്ഥികൾക്കും മികച്ച വിജയം കൈവരിക്കാൻ കഴിയട്ടെ എന്നാശംസിക്കുന്നു.

**എം.കെ. റഫീഖ്**  
**പ്രസിഡണ്ട്**  
**ജില്ലാ പഞ്ചായത്ത്, മലപ്പുറം**

നസീബ അസീസ്  
ചെയർ പേഴ്സൺ  
ആരോഗ്യ വിദ്യാഭ്യാസ സ്ഥിര സമിതി

ശ്രീമതി. സ്മ

***Prepared by:***

---

---

- ◆ SHABEER M MUNDEKAT      GHSS EZHUR
- ◆ SHIJU P B                      GHSS KUTTIPPURAM
- ◆ SREEKANTH K                 GJHSS NADUVATTOM
- ◆ Dr. SAJIN S CHANDRAN      KMGVHSS THAVANUR
- ◆ RADHIKA C V                 GHSS ATHAVANAD
- ◆ SIVAPRASAD K A              IKT HSS CHERUKULAMBA

---

# Contents

---

1	The Discipline of Computing	5
2	Data Representation and Boolean Algebra	6
3	Components of the Computer System	11
4	Principles of Programming & Problem Solving	15
5	Introduction to C++ Programming	17
6	Data Types and Operators	19
7	Control Statements	23
8	Arrays	26
9	String Handling and I/O Functions	29
10	Functions	32
11	Computer Networks	35
12	Internet and Mobile Computing	40

---

# The Discipline Of Computing

## Evolution of the Computing machine

### 1. Abacus

‘Abacus’ means calculating board.  
It is used for performing arithmetic operations.  
It works on the basis of the place value system.

### 2. Difference Engine

Difference engine was developed by Charles Babbage.  
It is a machine that could perform arithmetic calculations and print results automatically.

### 3. Analytical Engine

The analytical engine was developed by Charles Babbage.  
It was able to perform all general purpose computations.  
The engine had a ‘store’ (Memory) and a ‘mill’ (Processor)  
The instructions for the machine were written by Augusta Ada King.

## Generations of Computers

Criteria	Generation				
	First	Second	Third	Fourth	Fifth
Technology	Vacuum Tube	Transistor	Integrated Circuit	Microprocessor	Artificial Intelligence
Operating System	None	None	Yes	Yes	Yes
Language	Machine	Assembly	High Level	High Level	High Level
Period	1940-1956	1956-1963	1964-1971	1971-Present	Present and Yet to come

## Questions

1. Difference engine was developed by -----.
2. Who is the first programmer in the world?
3. Expand ENIAC.
4. Prepare short note about the generations of computers.
5. Compare difference engine and analytical engine.

## Data Representation and Boolean Algebra

### Number System

The number of symbols used in a number system is called base or radix.

#### Decimal Number system

Base - 10  
Contain the digits 0 -9

#### Binary Number System

Base – 2  
Contain the digits 0 and 1.  
Each digit of a binary number is called bit (binary digit).

#### Octal Number System

Base – 8  
Contain the digits 0-7

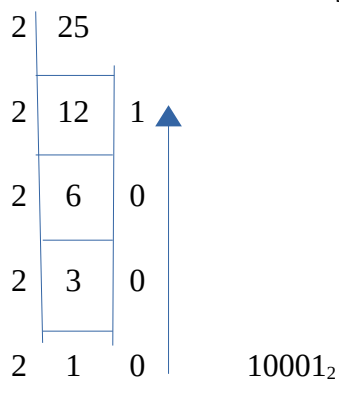
#### Hexadecimal Number System

Base – 16  
Contain digits 0-15  
The symbols A,B,C,D,E and F are used to represent the decimal numbers 10,11,12,13,14,15 respectively.

### Number conversion

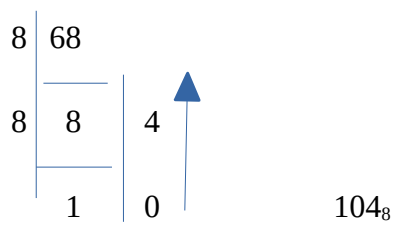
#### 1. Decimal to binary conversion

Divide the decimal number by 2 repeatedly, and take the remainders in reverse order.



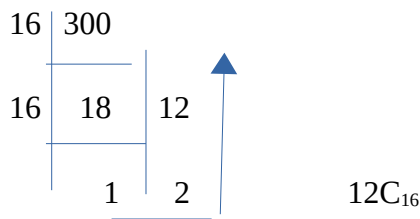
#### 2. Decimal to octal conversion

Divide the decimal number by 8, and take the remainders in reverse order.



### 3. Decimal to hexadecimal conversion

Divide the decimal number by 16, and take the remainders in reverse order.



### 4. Binary to Decimal conversion

Multiply each digit by place value ( $2^0, 2^1, 2^2, \dots$ ) and find the sum.

$$1011_2 = 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 = 8 + 0 + 2 + 1 = 11_{10}$$

$2^3$	$2^2$	$2^1$	$2^0$
1	0	1	1

### 5. Octal to Decimal conversion

Multiply each digit by place value ( $8^0, 8^1, 8^2, \dots$ ) and find the sum.

$$142_8 = 1 \times 8^2 + 4 \times 8^1 + 2 \times 8^0 = 64 + 32 + 2 = 98_{10}$$

### 6. Hexadecimal to Decimal conversion

Multiply each digit by place value ( $16^0, 16^1, 16^2, \dots$ ) and find the sum.

$$201_{16} = 2 \times 16^2 + 0 \times 16^1 + 1 \times 16^0 = 512 + 1 = 513_{10}$$

### Short cut conversions

1. Octal to binary
2. Binary to octal

The following table can be used for Octal to binary conversion and binary to octal conversion.

Octal	0	1	2	3	4	5	6	7
Binary	000	001	010	011	100	101	110	111

$$100001_2 = 41_8$$

3. Hexadecimal to binary
4. Binary to Hexadecimal

The following table can be used for hexadecimal to binary and binary to hexadecimal.

Binary	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
Hexa decimal	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111

$$89_{16} = 10001001_2$$

## Representation of Integers

Three methods for representing an integer number in computer memory.

### Sign and magnitude representation

Positive numbers – Leftmost bit is 0

Negative numbers – Leftmost bit is 1.

E.g. To represent -32 in sign magnitude form,

--- Find the binary equivalent of 32 --- > 100000

--- Make the binary equivalent 8 bit form ---> 00100000

--- Change the leftmost bit to 1 ---> 10100000

### 1's complement representation

Positive numbers - represented as it is.

Negative numbers - represented as 1's complement of the corresponding number.

(1's complement can be obtained by interchanging the 1's and 0's).

E.g. To represent -27 in 1's complement form

--- Find the binary equivalent of 27 ---> 11011

--- Make it to 8 bit form ---> 00011011

--- Find the 1's complement ---> 11100100

### 2's complement representation

Positive numbers - represented as it is.

Negative numbers – represented as 2's complement of the corresponding number.

(2's complement can be obtained by adding 1 to 1's complement)

E.g. To represent -20 in 2's complement form

--- Find the binary equivalent of 20 ---> 10100

--- Make it to 8 bit form ---> 00010100

--- Find the 1's complement ---> 11101011 +

$$\begin{array}{r} 1 \\ \hline 11101011 \\ \hline \end{array}$$

## Representation of characters

### **ASCII**

- American Standard Code for Information Interchange
- ASCII-7 uses 7 bits for representation
- ASCII-8 uses 8 bits for representation

### **Unicode**

- Developed by Unicode consortium
- Can represent all written languages in the world.



## Boolean Algebra

**Boolean algebra** deals with the values 'true' and 'false', denoted as 1 and 0. The name Boolean is given to honour George Bool, British Mathematician.

### Boolean operators and logic gates

#### Boolean operators or logical operators –

To perform operations on boolean values. The operands will be 0 or 1.

Basic logical operators are :

OR – Logical Addition

AND – Logical Multiplication

NOT – Logical Negation

#### Logic gates

Computers perform logical operations with electronic circuits, called logic circuits. A Logic circuit is constructed with logic gates.

A **logic gate** is a physical device that can perform logical operations on one or more logical inputs and produce a single output.

Three logic gates are : AND, OR, NOT.

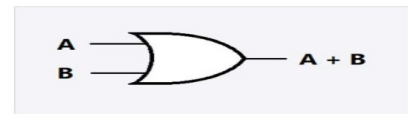
**Truth table** is a table that shows all possible inputs and their corresponding outputs for the given operation.

#### OR gate

- OR operator is used for logical addition.

- The symbol used for OR operation is +. The expression  $A+B$  is read as A OR B.

The logic gate used to implement the logical OR operation is called **logical OR gate**.



Truth table for OR operation

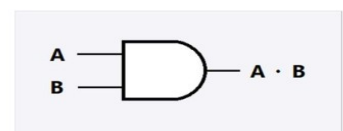
A	B	A + B
0	0	0
0	1	1
1	0	1
1	1	1

#### AND gate

- AND operator is used for logical multiplication.

- The symbol used for AND operation is . (dot).

The expression  $A.B$  is read as A AND B.

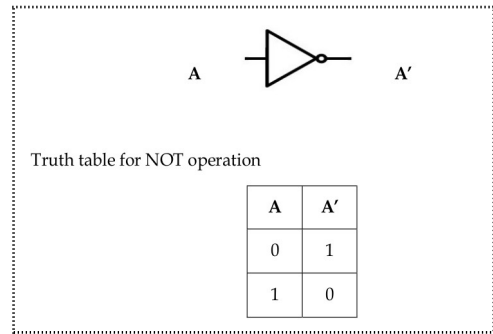


Truth table for AND operation

A	B	A . B
0	0	0
0	1	0
1	0	0
1	1	1

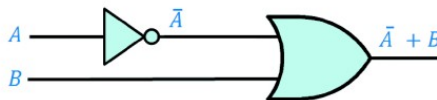
## NOT gate

- NOT operator performs logical negation.
- It requires only one operand.
- The symbol used for this operation is  $\bar{A}$  or  $A'$
- NOT gate is also known as inverter.



## Circuit Design for simple Boolean Expressions

Consider the expression  $\bar{A} + B$



## Questions

1. Explain any three number systems.
2. Represent -53 in all the three forms.
3. Draw the logic gate and truth table of AND operation.
4. Convert  $101010_2$  to Octal and Hexadecimal number systems.
5. Write short notes on :
  - (a) ASCII
  - (b) Unicode

## Components of Computer System

### I. Processor

- A processor is also called **CPU or Central Processing Unit**.
- A processor does **two main functions**:
  - *Does all the computing and decision making operations.*
  - *Coordinates the working of a computer.*
- The commonly used processors are Intel core i3, core i5, core i7, AMD Quadcore etc.

### Registers

- Registers are temporary storage locations inside the CPU for accessing the instructions and datas more quickly than that in the memory.
- Important registers in a CPU are:
  - **Accumulator**
  - **Memory Address Register(MAR)**
  - **Memory Buffer Register(MBR)**
  - **Instruction Register(IR)**
  - **Program Counter(PC)**

### II. Memory

- Memory is a place where the data, instructions and results are stored temporarily or permanently.
- Let us learn some measuring units:

Binary Digit =1 bit

1MB (Mega Byte) =1024 KB

1 Nibble = 4 Bits

1GB (Giga Byte) =1024 MB

1 Byte = 8 bits

1TB (Tera Byte) = 1024 GB

1KB (Kilo Byte) =1024 Bytes

1PB (Peta Byte) = 1024 TB

### Random Access Memory(RAM)

- RAM is a primary memory that the processor can read from and write into.
- It stores the operating system, application programs and data when the computer is functioning.
- It allows reading and writing.
- It is volatile, ie its contents are lost when the device is powered off.

### **III. e-Waste**

- *e-Waste or Electronic waste* may be defined as discarded computers, office electronic equipment, entertainment devices, mobile phones, television sets and refrigerators.

#### **e-Waste disposal methods**

The different methods for disposing e-Waste are:

##### **a. Reuse**

- Second-hand use or usage after the equipment has been upgraded or modified.
- It reduces the volume of e-Waste generation.

##### **b. Incineration**

- Controlled and complete combustion process in which the waste is burned in specifically burned incinerators at a high temperature in the range of 900 to 1000 degree Celsius.

##### **c. Recycling of e-Waste**

- It is the process of making or manufacturing of new products from a product that has originally served its purpose.

##### **d. Landfilling**

- Widely used, but not a recommended method is called Landfilling.

### **IV. System Software**

- System Software is a set of system programs which helps in the execution of general user's computational requirements on a computer system.
- The components of a system software are:

#### **a. Operating System**

#### **b. Language Processors**

#### **c. Utility software**

##### **a. Operating system**

- It is a set of programs that acts as an interface between the user and the computer hardware.

*Main objective:* To make the computer convenient to use.

#### **Features of OS**

- Provides an environment for user to execute programs.
- Helps to use the computer hardware in an efficient manner.
- Controls and Co-ordinates the operations of a computer.
- It is the first program that is loaded from the hard disk in the computer and resides in the memory till the system is shutdown.
- It acts as a resource manager.
- It tries to prevent errors and improper use of the computer.

## **Functions of an Operating System**

- i. Process management
- ii. Memory management
- iii. File management
- iv. Device management

## **b. Language Processors**

- Inorder to communicate with the computer, the user needs a language that is understood by the computer.
- Language processors are the system programs that translate programs written in high level language or assembly language into its equivalent machine language
- So computer languages are classified into 2 types.

### **a. High level languages(HLL)**

### **b. Low level languages(LL)**

#### **i. Machine language**

#### **ii. Assembly language**

### **a. High level languages(HLL)**

- These are English languages.
- Simpler to understand than low-level languages.
- Not understandable to the computer.
- Example: C++.

### **b Low level languages(LL)**

- Called as machine-oriented languages. There are two types.

#### **i. Machine language**

- The language which uses binary digits (0 and 1) is called machine language.
- Writing a program in machine language is very difficult.

#### **ii. Assembly language**

- It is an intermediate-level programming language.
- It is easier to write programs as compared to machine language.

## **Need for a language processor**

- To convert the programs written in high level language which are not understood by the computer to a low-level language.

**Types of language processors :-** There are three types of language processors.

### **a. Assembler**

### **b. Interpreter**

### **c. Compiler**

- **Assembler**

It translates the program code written in assembly language into machine language.

- **Interpreter**

It translates the program code written in high-level language into machine language line by line.

- **Compiler**

It translates the program code written in high-level language into machine language completely.

## **V. Free and Open Source Software**

It gives the user the freedom to use, copy, distribute, examine, change and improve the software.

### **Freedoms defined by FSF (Free Software Foundation)**

Freedom 0 - The freedom to run program for any purpose.

Freedom 1 - The freedom to study how the program works and the provision of accessing the source code.

Freedom 2 - The freedom to distribute copies of the software. Freedom 3 - The freedom to improve the program and release your improvements to the public.

### **Examples of Free and Open Source Software**

- a. **GNU/Linux:** It is a computer operating system
- b. **GIMP:** It is an image editing software.
- c. **Mozilla Firefox:** It is one of the most popular web browsers.
- d. **OpenOffice.org:** It is a complete office suite that contains word processor (Writer).

### **Questions**

1. 1 Byte = ----- bits.
2. Which among the following is NOT a recommended method for e-waste disposal?  
a) Reuse      b) Incineration      c) Land filling      d) Recycling
3. Which one of the following is NOT a free and open source software?  
a) GNU/ Linux      b) MS Office  
c) GIMP      d) Mozilla Firefox
4. Write the names of any four CPU registers.
5. List four major functions of operating system.
6. Define the term e-Waste. Explain briefly about any two E-waste disposal methods.
7. List and explain various components of System software.

# Principles of Programming & Problem Solving

## Phases (Steps) in developing a program




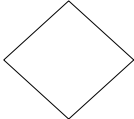

Programming is the act of giving instructions precisely to a computer to solve a problem. The following are major steps.

- i. **Problem Identification**
- ii. Preparing **algorithms and flowcharts**
- iii. **Coding** the program using programming language
- iv. **Translation**
- v. **Debugging**
- vi. **Execution and Testing**
- vii. **Documentation**

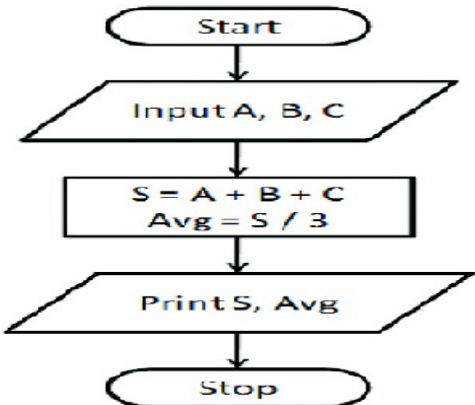
**Algorithm :-** It is a set of instructions written to solve a problem.

## Flowchart :-

It is the pictorial representation of an algorithm. The following are the various symbols used in a flowchart.

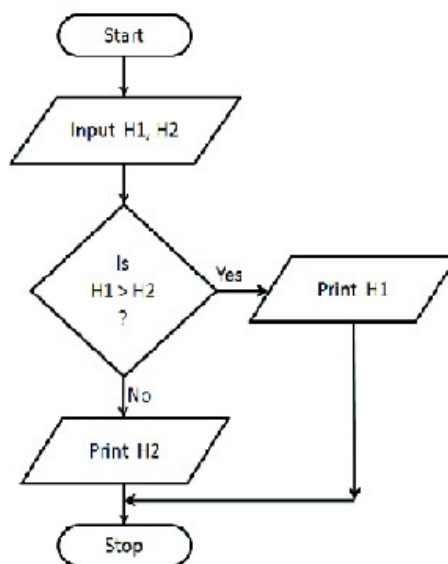
1. Terminal (To show start or Stop) 
2. Parallelogram (Input/Output) 
3. Rectangle (Process) 
4. Rhombus (Decision) 
5. Arrow or Flow lines Four Directional Arrow Marks
6. Circle (Connector) 

## Algorithm & Flowchart to find sum and average

<p>Step 1: Start                  Step 2: Input A,B,C                  Step 3: <math>S = A+B+C</math>                  Step 4: <math>Avg = S/3</math>                  Step 5: Print S, Avg                  Step 6: Stop</p>	 <pre> graph TD     Start([Start]) --&gt; Input[/Input A, B, C/]     Input --&gt; Process[S = A + B + C Avg = S / 3]     Process --&gt; Output[/Print S, Avg/]     Output --&gt; Stop([Stop])                 </pre>
---	--

## Algorithm to find taller (largest) height among two students

Step 1: Start  
Step 2: Input H1, H2  
Step 3: If H1>H2 Then  
Step 4: Print H1  
Step 5: Else  
Step 6: Print H2  
Step 7: End of If  
Step 8: Stop



## Debugging

Debugging means detecting and correcting errors in a program. Errors in a program are called bugs and detecting and correcting errors is called debugging. There are three types of errors in a program.

a. Syntax error :- It is the error committed by violating rules in a programming language.

Eg; Missing a semicolon in a line

b. Logical error :- Here, program gives you a wrong result for the input you gave.

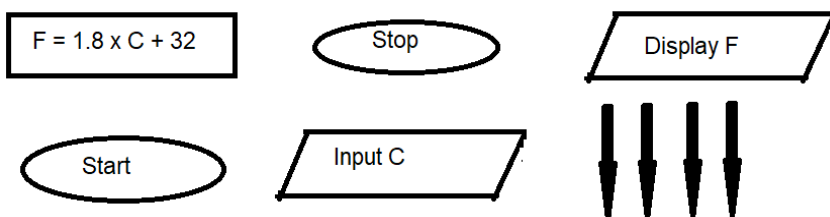
Eg; Using \* instead of +

c. Run-time error :- It is the error which interrupts a program.

Eg; Division by 0

## Questions

1. ----- is the symbol used in a flowchart to show a process.
2. The process of detecting and correcting errors is called -----
3. What is debugging? Explain various types of errors.
4. Rearrange the following stages of program development in correct sequence.  
Debugging, Problem Identification, Execution and Testing, Coding
5. List various phases in programming.
6. Write an algorithm to display sum of three numbers.
7. Draw a flowchart to display smallest of two numbers.
8. Given below are symbols related to a flowchart. Arrange them in correct sequence.





# Introduction to C++ Programming

## Tokens

A token is a word in C++. They are the fundamental building blocks of the program. They are also called **lexical units**. The following are various tokens in C++.

1. Keywords
2. Identifiers
3. Literals
4. Punctuators
5. Operators

### 1. Keywords: -

It is a word with a special meaning in C++. Their meaning cannot be changed. They must be in small letters.

Eg; *int, float, double, for, if, switch, break, continue, char, void*

### 2. Identifiers: -

It is the user-defined word to name variables, labels and functions. The following rules must be followed in naming an identifier.

- a. It must begin with alphabet (a-z, A-Z) or underscore ( \_ )
- b. It may contain alphabets, digits or underscore
- c. White spaces (Tab, Space etc.) and special characters (@,%,\$ etc.) not allowed.
- d. It should not be a keyword. (ie int, float etc not possible).

Eg; total      mark2      a5                      name\_of\_student    axb                      etc. are valid identifiers

+A              2K              grand total              a\*b                      etc. are invalid identifiers

### 3. Literals:-

It is the data item that never change during running program. For eg; 2, 3.14, 'A', "Hello" etc. The following are major types of literals.

- a. Integer Literal:- It is a number without decimal point. Eg; 125, -65, 7895
- b. Floating Point Literal:- It is a number including decimal point. Eg; 2.75, -100.658
- c. Character Literal:- Any character enclosed in a pair of single quotes. Eg. 'A', '+', '9'
- d. String Literal:- It is a sequence of one or more characters enclosed in pair of double quotes.  
Eg; "Nandana", "Welcome to C++", "9", "A", "int"

#### 4. Punctuators

They are symbols used in between statements in C++ such as spaces, coma (,), semicolon (;) etc.

#### 5. Operators

It is a symbol that tells about a specific kind of operation.

For eg; + for addition, - for subtraction, \* for multiplication etc.

#### Questions

1. What is a token in C++? List out any four tokens.
2. Which of the following is a valid identifier?  
a. A+B      b. A-B      c. AxB      d. None of these
3. Differentiate the following tokens in C++.  
A      'A'      "A"
4. What are the different rules for naming an identifier?
5. List out various types of literals with suitable examples.

# Datatypes and Operators

## Data types

It is used to identify nature and type of data stored in a variable.

### Fundamental data types

Data type	Type of value	Memory (bytes)	Eg:
void	null or empty data	0	
char	character values	1	'A', '\n'
int	integer values	4	84, -4
float	real values	4	5.6, -89.5
double	Real values(more precision than float)	8	

Integral datatypes

Floating point datatypes

Numerical datatypes

**Variables:** Names given to memory locations.

Eg: int Num=18;

1001

18
----

Num

- i. Variable name : The name of the variable. (**Num**)
- ii. Memory address(L-value):The memory address.(**1001**)
- iii. Content(R-value) :The value stored in the variable.(**18**)

**Operators:** Symbols that indicate an operation. Eg: +, <, \*, &&

In the expression  $a+b$ , + is the **operator** and a and b are the **operands**.

## Classification of operators

### (1)Based on number of operands:

Category	No. of operands	Example
Unary	1	Unary +, unary -, ++,--
Binary	2	+, &&, <
Ternary	3	?:

### 2. Based on nature of operation

(a) Arithmetic operators : Used for arithmetic operations such as addition(+), subtraction(-), multiplication(\*), division(/) and modulus (%).

x	y	x+y	x-y	x*y	x/y	x%y
7	3	10	4	21	2	1 (remainder)

## **(b) Relational operators**

Used for comparing numeric data, resulting in either **true** (1) or **false** (0). The relational operators are <(less than), >(greater than), <=(less than or equal to), >=(greater than or equal to), ==(equal to), !=(not equal to).

x	y	x<y	x>y	x<=y	x>=y	x==y	x!=y
7	3	0	1	0	1	0	1

## **(c) Logical operators**

Used to combine logical values, resulting in true or false. The logical operators are &&(AND), || (OR) and ! (NOT)

E1	E2	E1&&E2	E1  E2
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	1

E	!(E)
0	1
1	0

Eg: (2>5) && (4<6) results in 0

(2>5) || (4<6) results in 1

!(2>5) results in 1

## **Input / Output operators:**

The **get from (extraction)** operator (>>) is used for input. Eg: cin>>a;

The **put to(insertion)** operator (<<) is used for output. Eg: cout<<a;

## **Assignment operator (=)**

Used to store a value in a variable. Eg: a=5;

## **Difference between = and == operators.**

=	==
Assignment operator	Relational operator
Assigns value to a variable	Compares values

## **Expressions**

It is a Combination of operators and operands.

**(a) Arithmetic expressions:** contains arithmetic operators. Eg: m + n \* y

(i) **Integer expressions:** contains integer operands.

(ii) **Floating point(Real) expressions:** contains real operands.

(iii) **Constant expressions:** contains constant values. (eg: 5+m/3)

**(b) Relational expressions:** contains relational operators. Eg: x > y

**(c) Logical expressions** :contains logical operators. Eg: a<b && c>d

## Statements

Smallest executable unit of a program. *C++ statement ends with;*

**1. Declaration statement:** Specifies the type of data that will be stored in a variable.

*Syntax: datatype var\_name;*

eg: float avg;

**2. Assignment statements:** It stores a value to a variable. Eg: a=15;

**3. Input statements:** Specifies an input operation. Eg: cin>>score;

**4. Output statement:** Specifies output operation. Eg: cout<<score;

The multiple use of input/output operators in a single statement is called **cascading of I/O operators**.

Eg: cin>>x>>y>>z;

cout << "Sum=" << sum << "\n";

## Basic structure of a C++ Program

```
#include <headerfile>      ----> line 1
```

```
using namespace identifier; ----> line 2
```

```
int main()                  ----> line 3
```

```
{ statements;
```

```
:
```

```
return 0; }
```

**Line 1:** Preprocessor directive :

Instructs the compiler to perform an action before actual compilation. Starts with the symbol #.

eg: #include<iostream> --instruction to link the header file iostream.

**Line 2:** namespace statement:

Tells the compiler to use namespace **std** in this program. std is the standard namespace in which a lot of objects, including cin and cout are defined.

**Line 3:** Function header:

main() is the essential function for every C++ program. A c++ program starts and ends within the main() function.

## A sample program

```
#include<iostream>
using namespace std;
int main()
{
    cout<<"Hello, Welcome to C++";
    return 0;
}
```

## Questions

1. The \_\_\_\_ datatype is used for null or empty set of values.
2. Which among the following is the insertion operator?  
( << , >> , < , > )
3. The fundamental datatype that uses 8 bytes of memory is \_\_\_\_.
4. Predict the output of the following expressions if m=7 and n=2  
(i) m/y (ii) m%y (iii) (m==n) (iv) (m>10)&&(n<5)
5. Distinguish between == and = operators.
6. What is the importance of datatype? Name two integral datatypes in C++.
7. Find the Rvalue and memory size of the variable total from the following c++ statement  
float total = 34.6;
8. How is a binary operator different from a unary operator? Give one example each.
9. Write a C++ statement to declare a variable to store the mark of a student.
10. What are statements in C++? Explain any three types of statements with an example each.

## Control Statements

### Control Statements:

The statements used to change the normal sequential flow of program execution.

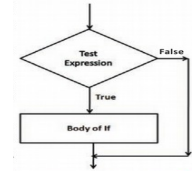
Two types: (1) Decision (2) Iteration

**1. Decision(Selection) Statements** : Statements are selected for execution based on a condition.

(a) **if**

Syntax: if (condition)  
 { statement block ; }

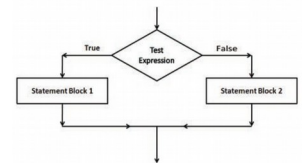
eg: if(mark>35)  
 { cout<<"Passed"; }



(b) **if-else**

Syntax: if (condition)  
 {statement block 1;}  
 else  
 {statement block 2;}

eg: if(mark>35)  
 {cout<<"Passed";}  
 else  
 {cout<<"Failed";}



(c) **if-else if ladder**

Syntax:if (condition 1)  
 statement block 1;  
 else if (condition 2)  
 statement block 2;  
 .....  
 else  
 statement block n;

eg: if (score >= 80)  
 cout<<"A Grade";  
 else if (score >= 60)  
 cout<<"B Grade ";  
 else if (score >= 40)  
 cout<<"C grade";  
 else  
 cout<<"D grade";

(d) **switch**

Syntax: switch (expression)  
 {  
 case constant\_1: statement block 1;break;  
 case constant\_2: statement block 2;break;  
 :  
 default : statement block n;  
 }

eg: switch (game)  
 {  
 case 1: cout<<"Football"; break;  
 case 2: cout<<"Cricket"; break;  
 default: cout<<"Invalid ";  
 }

All switch statements can be converted to else if ladder. But not all if-else if can be converted to switch.

### Difference between switch and if else if

switch	if else if ladder
Permits multiple branching.	Permits multiple branching.
Can be used only for checking equality	Any relational or logical condition can be checked
Case constant is integer or character	Can compare against a set of values including floating numbers
Uses <b>default</b> when no match is found, and <b>break</b> is used to exit.	When no condition is true, else block executes.

## 2. Iteration(Looping) statements

Statements that allow repeated execution of a set of one or more statements.

**There are 3 loops in C++** : while, for, do-while

A loop has the following components:

- Loop components** –
- (1) initialisation - statement that gives starting value to loop variable (**i=1**)
  - (2) condition – the test expression. (**i<=10**)
  - (3) updation – statement that changes the value in loop variable (**i++**)
  - (4) body of loop- set of statements to execute repeatedly (**cout<<i;**)

### for loop

```
for (i=1;i<=10;i++)
{ cout << i; }
```

### while loop

```
i=1;
while(i<=10)
{cout<<i;
i++; }
```

### do while loop

```
i=1;
do
{cout<<i;
i++;
}while(i<=10);
```



Here, **i** is the loop variable (its value controls the loop).

[All the three loops given above produce the same output – displays numbers from 1 to 10]

### Entry controlled loop

1. condition before loop body
2. loop will run only if condition is true

eg: for, while

### Exit controlled loop

1. condition after loop body
2. loop will run atleast once even if condition is true/false

eg: do while

## Jump Statements

Statements that facilitate the transfer of program control from one place to another. C++ provides four types of jump statements - break, continue, goto, return. Two of them are explained below:

### break statement

Takes the program control outside the immediate enclosing loop ( for , while , do...while ) or switch statement.

### continue statement

Used for skipping over a part of the code within the loop-body and forcing the next iteration.

<b>break</b>	<b>continue</b>
Forces termination of the loop	forces next iteration of the loop.
Used with switch and loops.	Used only with loops.



## Questions

- 1) Write the syntax of if else statement.
- 2) Compare entry controlled loop and exit controlled loop.
- 3) ----- statement takes the program control out of loop even though the test expression is true.
- 4) Differentiate 'break' and 'continue' statement in table form
- 5) Write the syntax of 'for' statement used in C++.
- 6) Control Statements can be classified in to two. Explain each with example.
- 7) Explain loop elements with a suitable example
- 8) "There are situations where if ..else if ladder cannot be rewritten using switch statement ". What are the situations?
- 9) Do as instructed :

(a) Rewrite using switch

```
if(ch==1) cout<<"Red";  
else if (ch==2) cout<<"Blue";  
else if (ch==3) cout<<"Green";  
else cout<<"Wrong code";
```

(b) Rewrite using do while and identify the loop components.

```
for(a=5; a<=20; a++)  
{ cout<<a*a ; }
```

**Array**

An Array is a collection of variables of same data type that are referenced by a common name. Another definition “An array is a collection of homogeneous elements that are referenced by a common name”.

**Advantages of arrays**

If we want to write a program which can accept marks of 60 students of a class, calculate their average etc. If we solve this problem by using variables, we need to declare 60 variables to store the different marks. Managing all these variables is not an easy task and it will make the problem complex and program is not understandable.

But if we declare an array to store these marks, we need to manage only one variable, the array name.

1. It is easy to represent and manipulate array variables.
2. Array use a contiguous memory locations
3. Readability of a program increased.

**Declaration of Arrays**

An array can be declared as **data type array name [size];**

Eg :- `int mark[5];`

Here mark is the name of the array which can hold 5 values of integer data type. All arrays consist of contiguous memory locations.

Mark[0]      mark[1]      mark[2]      mark[3]      mark[4]

55	48	36	58	45
4 bytes	4 bytes	4 bytes	4 bytes	4 bytes

The elements of an array can be accessed by its **subscript or index value**. The elements of arrays will be referred to as **arrayname [index];**

The index value (subscript) always starts from 0 (zero).

- ie the first value (say 55)in the array mark will be referred as      mark[0]
- second value (say 48) in the array mark will be referred as      mark[1]
- third value (say 36)in the array mark will be referred as      mark[2]
- fourth value (say 58)in the array mark will be referred as      mark[3]
- fifth value (say 45)in the array mark will be referred as      mark[4]

Consider the array declaration, `int mark[20];`. Then how much memory can be allocated by this array?

We know every integer occupy 4 bytes for storage in memory. Then `mark[20]` can hold 20 integer values in the array. So  $20 \times 4 = 80$  bytes of memory will be occupied by the array `mark[20]`.

- An array is a derived data type,
- The data type of array elements is known as the base type of the array.
- An array is a collection of consecutive memory locations referenced by a single variable.

## Types of Arrays

Arrays are of two types

- One dimensional array (Single dimensional Array)
- Multi Dimensional Array

## Initialization of Arrays

An array can be initialized in two ways

### 1) *Along with the declaration*

Eg :- `int mark [5] = { 55, 48, 36, 58, 45 } ;`

### 2) *direct Assignment*

Eg :- `mark[0] = 55; mark[1] = 48; mark[2] =36; mark [3] = 58 ; mark [4] = 45;`

## Accessing elements of array

We can access the elements of array with the help of any loop. The loop control variable determines the index value of array. Each element is specified by array name with the subscript.

## Array operations

- **Traversal**- Moving through the array elements. Displaying array elements is an example of traversal
- **Searching**- Finding the location of the given element in the array. Two methods (Linear Search, Binary Search)
- **insertion**-Inserting a new element into array.
- **deletion**-deleting an element from array
- **sorting**- Arranging the array elements in some logical order (such as ascending or descending). Two methods selection sort, Bubble sort.
- **Merging** – Combining more than one arrays into single array

Note :- The following program explains how the array traversal operation is performed. We used for loop to perform the traversal. The following example also teaches how insertion and display of array elements can be performed.

Eg :- Write a program to enter 10 elements into an array and print it.

```
# include <iostream>
using namespace std;
int main( )
{
    int num[10], i;
    cout << " Enter elements in to the Array \n ";
    for ( i = 0 ; i <= 9; ++ i)
        cin >> num[i] ;
    cout << " \n The elements in the array are \n ";
    for ( i = 0 ; i <= 9; ++ i)
        cout << num [ i] << "\n";
    return 0;
}
```

In the above program the for loop is executed 10 times for loop control variable i, values ranging from 0 to 9. These values are array index values. After inputting, another for loop is used to access the values from the array for outputting.

## **Questions**

1. Illustrate the memory allocation for a single dimensional array in C++
2. Write a C++ program to input the scores of 5 students and display them in reverse order using an array
3. What is an array? Write C++ program to declare and use a single dimensional array for storing the computer science marks of all students in your class
4. Declare an array of size 5 and initialize it with numbers 8,7,2,4 and 6
5. Write a C++ program to illustrate array traversal

## String Handling And I/O Functions

### Array declaration for string and initialization

A string is declared using character array.

Eg: `char my_name[10];`

This character array can be initialized with characters as follows

`char my_name[10]={'N','i','k','e','t','h'};`

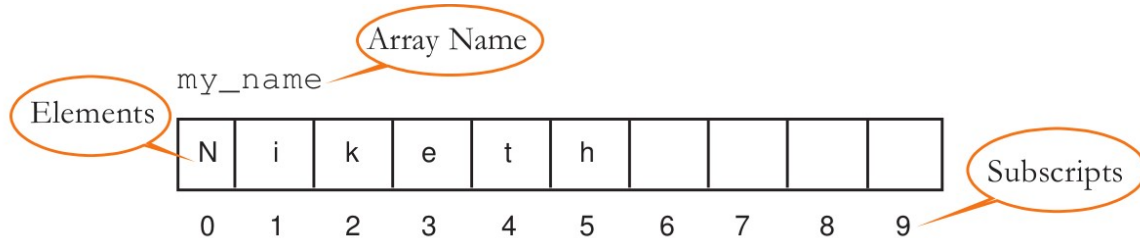


Fig. 9.1 : Memory allocation for the character array

In the above method 6 bytes are used to store the character array value.

If we input a string in a character array, the memory allocation is different

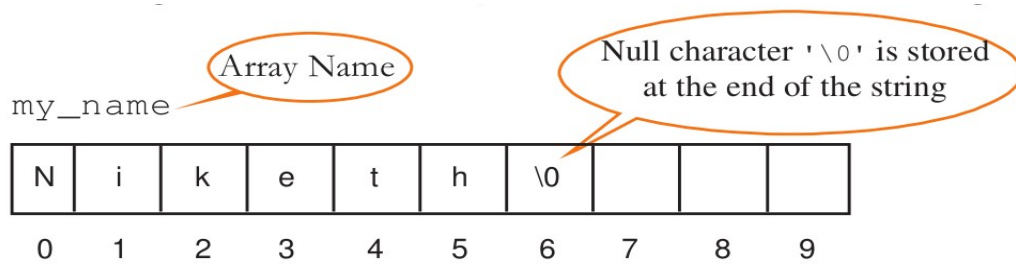


Fig. 9.2 : Memory allocation for the character array

In the above method 7 bytes are used to store the string value. The seventh byte is for strong Null character('\0' - End of string)

We can initialize a character array with a string as follows

`char my_name[10]="Niketh";`

### Input Output Functions

In C++ the input output operations are managed by a collection of functions and defined through header files namely 'iostream' and 'stdio'. The functions defined in 'stdio' are called console I/O functions and the functions defined in 'iostream' are called stream I/O functions.

#### gets() function

The function **gets()** is a console input function used to accept a string of characters including white spaces from the standard input device (keyboard) and store it in a character array.

The string variable (character array name) should be provided to this function as shown below:

`gets(character_array_name);`

## puts() function

The function **puts()** is a console output function used to display a string data on the standard output device (monitor). Its syntax is

```
puts(string_data);
```

The string constant or variable (character array name) to be displayed should be provided to this function.

Observe the following C++ code fragment:

```
char str[10] = "friends";  
puts("hello");  
puts(str);
```

The output of the code above will be as follows:

hello

We have discussed gets() function and its advantage in string input. There are some other console functions which allow character input. The following table shows these functions and their working of each.

Function	Example	Working
getchar()	ch=getchar();	The character input through the keyboard is stored in the variable ch.
getch()	ch=getch();	The typed (input) character is not be displayed on the screen, but is stored in the variable ch. The <Enter> key need not be pressed to delimit the data.

## Output Functions

We have discussed puts() function and its advantage in string output. There are some other console functions which allow character output

Function	Example	Working
putchar()	putchar(ch); putchar('A');	Displays the content of the character variable ch. The character 'A' is displayed.
putch()	putch(ch); putch('A');	Displays the content of the character variable ch. The character 'A' is displayed.

## Stream Functions

### Input functions

Function	Syntax	Use
<b>get()</b>	cin.get(ch); ch=cin.get(); cin.get(str,10); where str is character array.	A character variable should be provided to store the character. If a character array and size will be given instead of character variable, a string can be input.
<b>getline()</b>	cin.getline(str,len); cin.getline(str,len,ch);	Input function for a string. The delimiter will be the Enter key, the number of specified characters or the specified character.

## Output Functions

Function	Syntax	Use
<b>put()</b>	cout.put(ch); cout.put('A');	Displays the content of a character variable or a character constant provided within the parentheses.
<b>write()</b>	cout.write(str,len);	Displays string contained in the variable str of size len.

## Questions

1. Differentiate between put() and write() with an example.
2. How many bytes are required to store the following string?  
"HELLO WORLD"
3. Name two input stream functions in C++.
4. Which character is used to delimit the string in memory ?
5. What is the advantage of using gets ( ) function in C++ program to input string data? Explain with an example.

## CHAPTER 10

# Functions

### Concept of modular programming

In programming, the entire problem will be divided into small sub problems that can be solved by writing separate programs. This kind of approach is known as modular programming. The process of breaking large programs into smaller sub programs is called modularisation

### Merits of modular programming:

- Reduces the size of the program
- Less chances of error
- Reduces programming complexity
- Improves re usability

### Demerits of modular programming:

- Proper breaking down of the problem is a challenging task.
- Care should be taken while setting the hierarchy of the execution of the modules.

**Function:** function is a named unit of statements in a C ++ program to perform a specific task.

**Predefined functions or built-in functions:** The ready-to-use functions are built-in functions.

#### 1.String Functions

These are used for the manipulation of strings. We should include the header file cstring ( string.h in Turbo C++) in our C++ program to use these functions.

Function	Syntax	Operation
<b>strlen()</b>	int strlen(string);	To find the length of a string(Number of characters in a string)
<b>strcpy()</b>	strcpy(string1,string2);	To copy one string into another. Here string2 copied to string1.
<b>strcmp()</b>	strcmp(string1,string2);	To compare two strings, Case sensitive. Both strings are same returns 0.If string1 is alphabetically higher than string2 , returns a +ve value else returns a -ve value.
<b>strcmpi()</b>	strcmpi(string1,string2);	To compare two strings, Not case sensitive. Both strings are same returns 0.If string1 is alphabetically higher than string2 , returns a +ve value else returns a -ve value.
<b>strcat()</b>	strcat(string1, string2);	To append one string to another string. Here string2 is appended to string1.



## 2.Mathematical Functions:

Used for mathematical calculations. We should include the header file `cmath` ( `math.h` in Turbo C++ ) to use these functions in the program.

Function	Syntax	Operation
<code>abs()</code>	<code>int abs(int);</code>	To find the absolute value of an integer. Eg: <code>abs(-25)</code> is 25.
<code>fabs()</code>	<code>double fabs(double);</code>	To find the absolute value of a floating point number Eg: <code>fabs(-25.89)</code> is 25.89
<code>sqrt()</code>	<code>double sqrt(double);</code>	To find the square root of a number. Eg: <code>sqrt(100)</code> is 10.
<code>pow()</code>	<code>double pow(double,int);</code>	To find the power of a number. Eg: <code>pow(3,2)</code> is 3 <sup>2</sup> ie 9
<code>sin()</code>	<code>double sin(double);</code>	To find the sine value of an angle.
<code>cos()</code>	<code>double cos(double);</code>	To find the cosine value of an angle.

## 3.Character functions :

These functions are used to perform various operations of characters. The header file `cctype` ( `ctype.h` for Turbo C++) is to be included to use these functions in a program.

Function	Syntax	Operation
<code>isupper()</code>	<code>int isupper(char c);</code>	To check whether a character is in the upper case or not. Returns 1 if the given character is in the uppercase, and 0 otherwise.
<code>islower()</code>	<code>int islower(char c);</code>	To check whether a character is in the lower case or not. Returns 1 if the given character is in the lower case, and 0 otherwise.
<code>isalpha()</code>	<code>int isalpha(char c);</code>	To check whether the given character is an alphabet or not. Returns 1 if the given character is an alphabet, and 0 otherwise.
<code>isdigit()</code>	<code>int isdigit(char c);</code>	To check whether the given character is a digit or not. The function returns 1 if the given character is a digit, and 0 otherwise.
<code>isalnum()</code>	<code>int isalnum (char c);</code>	To check whether a character is an alphanumeric or not. The function returns 1 if the given character is an alphanumeric, and 0 otherwise.
<code>toupper()</code>	<code>char toupper(char c);</code>	To convert the given character into its uppercase. The function returns the upper case of the given character. If the given character is in the upper case, the output will be the same.
<code>tolower()</code>	<code>char tolower(char c);</code>	To convert the given character into its lower case. Returns the lower case of the given character.

## User defined functions

The function created by user is user defined function.

The syntax of a function definition is given below:

```
data_type function_name(argument_list)
{
    statements in the body;
}
```

The data\_type is any valid data type of C++. The function\_name is a user defined word (identifier). The argument\_list , which is optional, the body comprises C++ statements required to perform the task assigned to the function.

```
int sum(int a,int b)
{
    c=a+b;
}
```

**Arguments of functions:** Arguments or parameters are the means to pass values from the calling function to the called function. The variables used in the function definition as arguments are known as **formal arguments**. The constants, variables or expressions used in the function call are known as **actual (original) arguments**.

**Return value:** The return statement returns a value to the calling function and transfers the program control back to the calling function.

## Questions

1. List any four C++ mathematical functions.
2. Write the syntax and use of following functions In C++  
a) strcpy ( )      b) pow ( )      c) islower ( )
3. What do you mean by argument of a function? Explain formal arguments.
4. Describe the merits of modular programming.
5. List any three string functions in C++ and specify the value returned by them.
6. Differentiate between string functions strcmp( ) and strcmpi( ).

# Computer Networks

## Network

It is an interconnection of computers and other hardware devices like printers, scanners, etc using a communication medium.

### Advantages

1) **Resource sharing** - Any hardware /software resource in one system can be shared with other systems in the network.

2) **Price-performance ratio** - The cost of purchasing licensed software for each computer, can be reduced by purchasing network versions of such software in a network.

3) **Communication** - The computer network helps users to communicate with any other computer in the network through its services like e-mail, chatting, video conferencing, etc.

4) **Reliability**- In a network, it is possible to backup data on multiple computers. This helps users to retrieve data in the case of failures in accessing data.

5) **Scalability** - Computing capacity can be increased or decreased easily by adding or removing computers to the network.

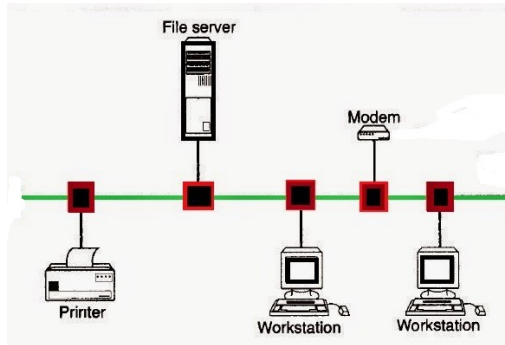


Fig 11.1 Computer Networks

### Some key terms

#### Bandwidth :

It measures the amount of data that can be sent over a specific connection in a given amount of time.

#### Noise:

It is unwanted electric or electromagnetic energy that lowers the quantity of data signals.

#### Node:

Any device which is directly connected to a network is called a Node.

### Data communication devices:

A data communication device provides an interface between computer and the communication channel.

#### 1) Switch:

- A switch is a device that connects several computers to form a network.
- It is an intelligent device, because it can transmit the received data to the destination only.
- It will store the addresses of all the devices connected to it .

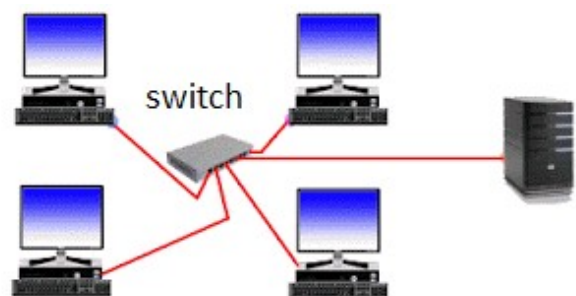


Fig 11.2 Switch

## 2) Bridge

- A bridge is a device used to segmentize a network.
- A network can be split into different segments and can be interconnected using a bridge.
- This reduces the amount of traffic on a network.

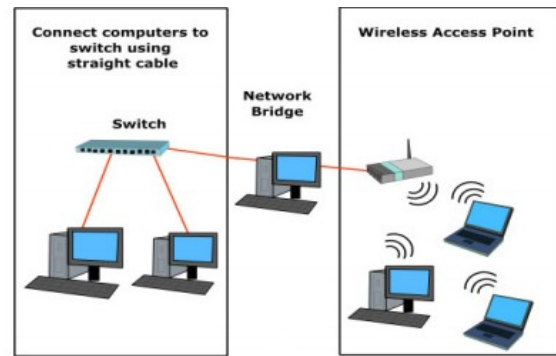


Fig 11.4 Bridge

## 3) Router

- A router is a device that can interconnect two networks of the same type using the same protocol.
- It can find the optimal path for data packets to travel and reduce the amount of traffic on a network.

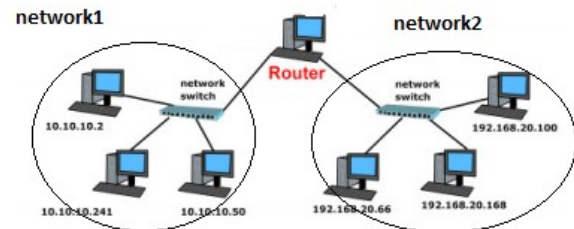


Fig 11.5 Router

## 4) Gateway :

- A gateway is a device that can interconnect two different networks having different protocols.
- It can translate one protocol to another.
- It can find the optimal path for packets to reach the destination.

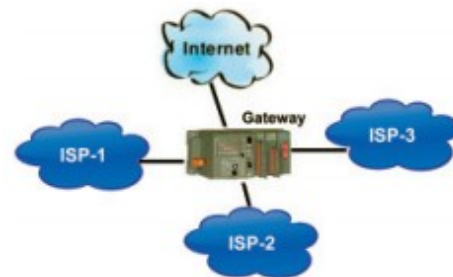


Fig 11.6 Gateway

**Modem:-** It is a **Data Terminal Equipment** that controls data flowing to or from a computer.

- A modem is a device used for communication between computers through telephone lines.
- The name is formed from **modulator and demodulator**.
- It converts digital signals received from a computer to analog signals for telephone lines and vice versa.

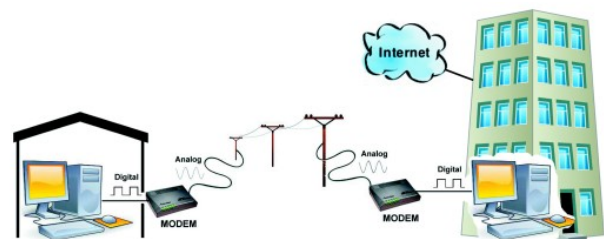


Fig 11.7 Modem

## Network topologies Topology :

The way in which the nodes are physically interconnected to form a network.

**Major topologies are bus, star, ring, and mesh.**

## 1) Bus topology:

- In bus topology, all the nodes are connected to a main cable called bus.
- A small device called a terminator is attached to each end of the bus.

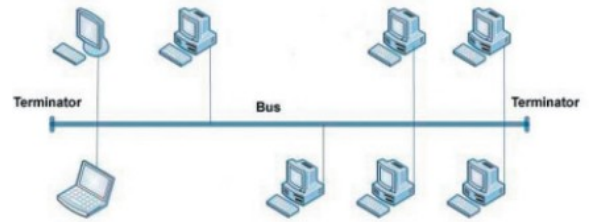


Fig 11.8 Bus topology

If a node has to send data to another node, it sends data to the bus. The signal travels through the bus and each node checks the bus and only the intended node will accept the data. When the signal reaches the end of the bus, the terminator absorbs the signal from the bus.

### Characteristics of bus topology

- 1) Easy to install.
- 2) Requires less cable length and hence it is cost-effective.
- 3) Failure of a node does not affect the network.
- 4) Failure of cable (bus) or terminator leads to a breakdown of the entire network.
- 5) Fault diagnosis is difficult.
- 6) Only one node can transmit data at a time.

## 2) Star topology

- In star topology, each node is directly connected to a hub/switch.
- If any node has to send some information to any other node, it sends the signal to the hub/switch.
- The signal is then broadcasted (in the case of a hub) to all the nodes but is accepted only by the intended node.
- In the case of a switch, the signal is sent only to the intended node.

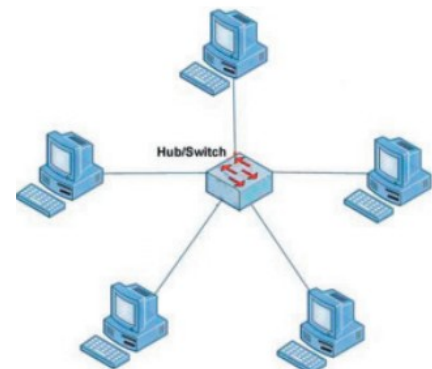


Fig 11.9 Star topology

### Characteristics of star topology

- 1) More efficient compared to bus topology.
- 2) Easy to install.
- 3) Easy to diagnose faults.
- 4) Easy to expand depending on the specifications of the central hub/switch.
- 5) Failure of hub/switch leads to failure of the entire network.
- 6) Requires more cable length compared to bus topology.

### 3) Ring topology

- In ring topology, all nodes are connected using a cable that loops the ring or circle.
- A ring topology is in the form of a circle.
- Data travels only in one direction in a ring.
- Each node regenerates the signal and passes to the next node until it reaches the intended node reaches.

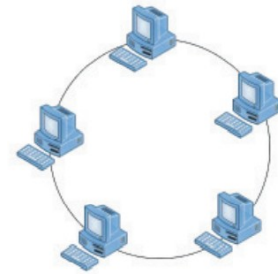


Fig 11.10 Ring topology

#### Characteristics of ring topology

- 1) No signal amplification is required as each node amplifies the signal.
- 2) Requires less cable length and hence is cost-effective.
- 3) If one node fails, the entire network will fail.
- 4) Addition of nodes to the network is difficult.

### 4) Mesh topology

- In mesh topology, every node is connected to other nodes.
- There will be more than one path between two nodes.
- If one path fails, the data will take another path and reach the destination.

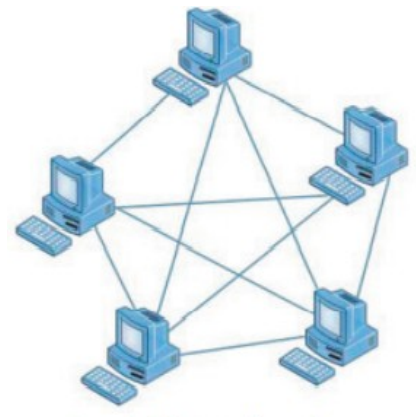


Fig 11.11 Mesh topology

#### Characteristics of mesh topology

- 1) Network will not fail even if one path between the nodes fails.
- 2) Expensive because of the extra cables needed.
- 3) Very complex and difficult to manage.

### Identification of Computers on a network

#### 1) Media Access Control (MAC) address:

- i. A universally unique address (12 digit hexadecimal number) assigned to each NIC (Network Interface Card) by its manufacturer.
- ii. MAC addresses are usually written in one of the following two formats:
- iii. MM : MM : MM : SS : SS : SS or MM – MM – MM – SS – SS – SS
- iv. The first half (MM:MM:MM) of a MAC address contains the ID number of the adapter manufacturer.
- v. The second half (SS:SS:SS) of a MAC address represents the serial number assigned to the adapter (NIC) by its manufacturer.

eg. 00:A0:C9 : 14:C8:35

#### 2) IP address:

- i. An IP address is a unique 4 part numeric address assigned to each node on a network, for their unique identification.
- ii. An IP address is a group of four bytes (or 32 bits) each of which can be a number from 0 to 255.

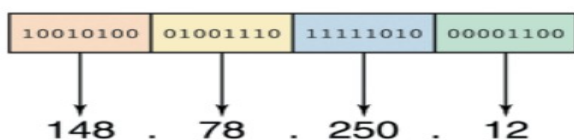


Fig 11.15 IP address

## Questions

- a) A----- is a computer peripheral that allows you to connect and communicate with other computers via telephone lines.
- b) In -----topology, all devices are connected to a central hub/switch.
- c) Explain any two advantages of computer networks.
- d) There are many advantages in using networked computers instead of stand-alone computers. Write any four advantages of them.
- e) Explain the functions of the network device **switch**.
- f) Define the term, topology. Consider that, your principal has decided to network your computer lab. Which topology will you suggest? Justify your answer.
- g) Compare ring topology and mesh topology.
- h) Define following terms related to computer network
- a) Bandwidth b) Noise c)Node

## Internet and Mobile Computing

### Internet Services :-

The following are major services in Internet.

1. World Wide Web (WWW)
2. Search Engines
3. Email
4. Social Media

### World Wide Web (WWW)

It consists of many servers storing documents containing text, images and videos. Every document is having a **URL (Uniform Resource Locator)** address. Clients can access the documents using hyperlink with the help of software called **browser**. The commonly used browsers are **Google Chrome, Mozilla Firefox, Opera, Internet Explorer, Safari** etc. Accessing documents using browsers is called **Web browsing**.

### Search engines

It is a special program designed to help people to find information available in WWW. It helps to list documents containing a specific keyword. The most popular search engines are **Google, Bing, Yahoo Search, Ask** etc.

### Email

It is the method of sending and receiving digital messages between computers using Internet. We can send document or picture or even video files as attachment along with a e-mail. The most popular email service providers are **gmail.com, hotmail.com** etc. Both the sender and receiver of email must have email address. The structure of a email address is given below along with an example.

user name@domain name                      [scertkerala@gmail.com](mailto:scertkerala@gmail.com)

An email contains To (Recipient Address), cc (Carbon Copy), Bcc (Blind Carbon Copy), Subject, Content and optionally attached files.

### Advantages

- i. Speed
- ii. Easy to use
- iii. Provision for attachment
- iv. Available anywhere any time.

### Limitations

- i. Email may carry viruses
- ii. Junk mails:- It means unwanted mails.



## Social Media

It refers to the use of mobile and web based technologies to communicate, create, share, discuss and modify content. It is a unreal world. The following are different types of social media.

- a. **Internet Forums**:- In this people can start or respond to a **thread**. A thread is a discussion topic. **Ubuntu Forum** is a popular forum.
- b. **Social blogs** :- A blog (web log) is an information website where entries arranged in reverse chronological order, ie most recent first. Single user and multiuser blogs are now available. Blogs are used for personal diaries, online advertising etc. **Blogger.com, Wordpress.com** are popular blogging sites.
- c. **Micro blogs**:- In this people exchange short sentences, images or video links about what they observe in their surroundings. Eg; **Twitter.com**
- d. **Wikis**:- Here, users can add or edit information in a webpage. Content will be closely monitored by a team of editors. Incorrect information and advertising etc. will be removed. Eg; **wikipedia.org**
- e. **Social networks**:- It allows to create personal web pages and share text, pictures, videos etc. Popular social networking sites are **facebook.com, linkedin.com** etc.
- f. **Content communities**:- It allows us to organize and share contents like photos, videos etc. **Youtube.com** is a popular video sharing site and **flickr.com** shares pictures.

### Advantages of social media

- i. Bring people together
- ii. Plan and organize events
- iii. Business promotion

### Limitations in use of social media

- i. Intrusion to privacy
- ii. Addiction
- iii. Spreading rumours

### Cyber security:-

The following are major threats in the use of computers and networks.

- a. **Computer Virus**:- It is a program attached to another program to spread. It affects normal operations of a computer. It may damage files in your hard disk. It is affected mostly through file sharing (USB drives, portable hard disk) or through email attachments.
- b. **Trojan Horse**:- It appears to be a useful program. But when it is installed it will be dangerous. A Trojan seriously damages a computer by deleting files. Some Trojans allows other users (hackers) to gather personal information from your computer.

- c. **Hacking:-** It is an effort to affect normal behavior of a network. Hacking done by computer security experts to detect faults is called **ethical hacking** and hackers are called '**white hats**'. Hacking done by computer criminals to steal information. Such criminals are called '**black hats**'.
- d. **Phishing:-** It is an attempt to collect usernames, password etc. by creating a duplicate website of banks and other institutions. Creating such misleading websites is called **spoofing**. People are forced to visit these websites through emails.

## **Questions**

1. Choose the odd one out.  
a. Yahoo      b. Gmail      c. Google      d. Bing
2. Name any two browsers familiar to you.
3. What are the limitations in the use of social media?
4. Define Virus and Trojan Horse.
5. Categorize the following into different social media types.  
(a) Youtube    (b) Wikipedia    (c) Twitter    (d) Facebook    (e) Ubuntu forum (f) blogger
6. List the names of various services in Internet. Explain any two of them.
7. What is email? Give the structure of an email address.
8. What is web browsing? List names of various software used for web browsing.
9. What are the advantages of using email?
10. Explain hacking. What do you mean by **white hats** and **black hats** hackers?
11. Which facility in Internet is used to lookup for web pages with specific keywords? List names of any three such services in Internet.