1. Central Processing unit (CPU) → it is also known as processor. CPU is the brain of the computer where all kind of processing is done. it works according to the set of instruction called programme.

The primary goal of CPU is to store the data temporarily in the register and perform arithematic and logical calculation.

— CPU has two parts —

(i) (ALU) — Arithematic & logical unit → It is responsible for carrying out Arithematic operation like addition, substraction, multiplication & division logical operation a>b, c≤d etc.
For this AND, OR, NOT etc. are used.

(ii) Control unit → Control unit the way data is moved between different components of the computer for this it generates signals. it control all the operation of the computer.

2. Memory unit → memory is known by that term storage. It can be classified as.
(i) Primary memory (ii) Secondary memory (iii) Cache memory

(i) Primary memory → it holds the data and Instruction which is being processed by the CPU. primary memory can be divided into —

(i) RAM (Random Access memory) → it holds the data and Instruction of currently running program. It is also called volatile memory. because when the power switched off all the data stored in RAM are watched.

RAM is of two types —

(i) SRAM (Static RAM)

(ii) DRAM (Dynamic RAM)

(ii) ROM (Read only memory) → it is also a part of primary memory it is also called Non-valatile memory. because when the power is switched off all the data stored in ROM are not watched.

ROM is BIOS (Basis Input output System)

— Types of ROM —

PROM → (Programmable ROM)

EPROM → (Erasable programmable ROM)

EEPROM → (Electrically Erasable programmable ROM)

(ii) Secondary memory → all such units which provide space for storing result and data permanently are called secondary storage devices.

Ex— Hard disk, CD, DVD, Pendrive etc.

(iii) Cache memory → Cache memory is very fast as compared to RAM. Frequently used data are kept in cache to improve the system performance.

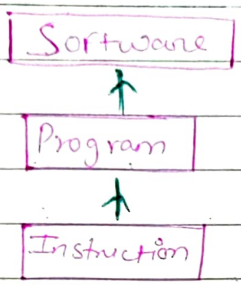
3. Input unit → Input devices are used to input the real world data into computer understandable form.

Ex→ Keyboard, mouse, scanner, barcode Reader, OMR, touch screen, MICR (Magnetic ink character Reader). etc.

4. Output unit → Computer send output in binary form which is then converted into human understandable form by the output device.

Ex - Monitor, Speaker, printer, projector, plotter etc.

Note :- Program → A set of instruction to do a particular task
Software is a collection of program which is used
to do a specific task.



— Structure of software —

Software is of two types :—

(1) System software (2) Application software

(1) System software → is used to make the computer hardware operational. It also provide platform to install other application software.

Example - System Software -

(1) Operating System → It manages and control Hardware and Software resources of a computer system.

E.g. windows, linux, Unix, Android etc.

it is primary system software.

(1) Operating System (ii) language translator.

Compiler and Interpreter are used as a translator to convert programming language intruction in to machine code (0,1)

(2) Application Software → are used to do a specific task for the End user. It is developed according to user requirement.

It can be of two types —

(i) Prewritten Software Pakages :- Ex- google, Chrome, Internet, Explorer, V/c, ms office, Adobe etc.

(ii) User written application program :- Ex- A software developed for a bank, college, railway etc.

→ Programming language → it is the computer language that is used by programmer to communicate with the computer.

Programming language provides set of rules / syntax / format in which various instruction are written. programming language can be classified as —

1. Machine language
2. Assembly language

(Low level language)

3. High level language
   - → Procedure language (C, C++, Java etc)
   - → Non procedural language (LISP, PROLOG etc)
   - → Problem oriented language (SQL, MAT, LAB etc)

• Machine language → This is the only language which computer understand directly.

In machine language the instruction are written in squence of zeros (0) & once (1) (0's & 1's) where zero represent off and 1 represent ON electrical state

- Advantage —

A processor can execute it without any translator.

- Disadvantage — (i) Machine dependent
  (ii) Error prone
  (iii) Difficult to debug and modify.

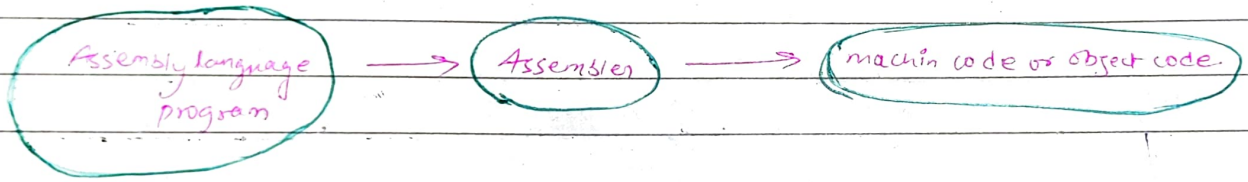
Assembly language :- Assembly language was the enhancement over machine language it uses mnemonics code and symbol.

Ex- A code ~~with~~ written in Assembly language.

```
ADD   R1 R2
MOV   R1 , R2
STORE  M[1002]
```

Computer cannot directly understand machine language instruction, So that a transmiter is Required. for this Assembler is used.

Assembler → it is a translator which converts Assembly language program into machine mode.



Advantage —

Improve Readability as compared to machine language.

Disadvantage —

1. Machine dependent
2. Difficult to debug

High Level language → are machine independent it is not required for the programmer to know the detail architecture of the processor, it uses basic english and mathematical Symbols.

Advantages —

- Readability → Easy to write program and understeind.
- Portability → A program can run on difficult machine.
- Debugging → Is fast and Easy
- Ease → In software development

Computer cannot directly understand the high level language program So, a translator is Required the Final compilar and Interpreter does it.

Compiler / Interpreter → Both are translator which converts high level language program into machine code.

High level language → Compiler/Interpreter → Machine code or Object code

— Difference between compiler & Interpreter —

Compiler	Interpreter
1. Scan the entire program before translating in into machine code.	— Translate and Execute the program line by line.
2. Convert entire program into machine code and execute only when all the syntax error are removed.	— It execute one line at a time after checking and correcting its syntax error and convert it into machine code.
3. Slow in debugging	— Fast in debugging.
4. Program execution time is less	— Program execution time is more.

Algorithm → An Algorithm is a finite set of Instructions followed to accomplish a task.
or
Step by step procedure to solve a problem.

— Characteristics of Algorithm —

1. Input: every algorithm must accept zero inputs.
2. Output: a algorithm should produce at least one output.
3. Finiteness: Algorithm should terminate after finite number of steps.

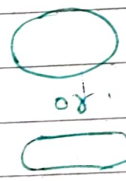
4. Definiteness: each instruction must be clear, well defined and precise.

5. effectiveness: Algorithm should be effective so, that it can carry out all the operation with less complexity.

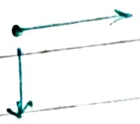
Ex. Write an Algorithm to Find Area of Rectangle.

Start
Step-1 Input l, b
Step-2 Area = l * b
Step-3 Print Area
END / STOP

Flow chart → Flow chart is a pictorial representation of sequence of operation necessary to solve a problem. Various symbols used to draw a flow chart are as follows.

Symbol	Name	Purpose
 or 	Terminal Box	Used for start and stop a flow chart.
	Process Box	Used for calculation and assigning value to a variable.
	Input/output Box	Used to input the value and to print the result.
	Decision Box	Used to test a condition, the result of the condition may be True or False.

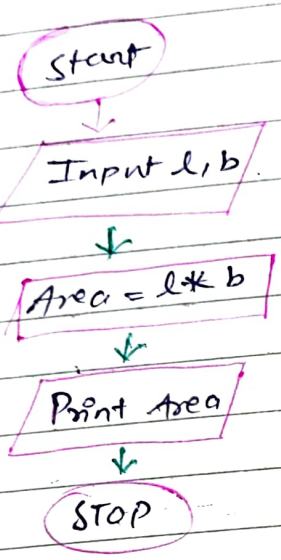
Flow line



Used to connect the Symbol, it shows the sequence of operation.

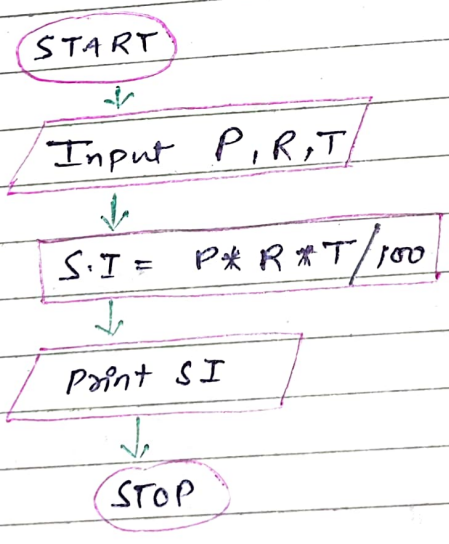
Que → Draw a flow chart to find the area of Rectangle.

Sol^n →



Que Draw flow chart to calculate simple interest.

Sol^n:



– 'C' Programming language —

- C programming language was developed by Dennis Ritchie at bell Lab USA in 1972.
- C language was developed to code Unix operating system
- C is a
  - General purpose
  - Structured
  - Portable
  - High level language
- C language was influenced by BCPL & B language
  (Basic Combined Programming Language)

– Application of 'c' language —

- It is known as a mother of all the programming language. Such as C++, C#, Java etc.
- Most of the operating System like windows, LINUX, UNIX are written in 'c' language.
- 'C' language is used to develop compilers, web browsers, drivers etc.

(1) Variable :-

- a memory location that store a value.
- that can change their value.

E.g: length, number, Height, Weight etc.

(2) Basic Data type :-

It indicates which type of value the variable will store.

(1) Int → Integer value (whole no.) // A number without decimal.

E.g. 5, -10, 350 etc

E.g. ~~int~~ int x; // x is a variable and the value of x would be integer.

(2) float → number with decimal

E.g. 5.35, 3.36, -5.5, 5.0 etc

E.g. float x; // x is a variable and value of x would be a float value.

(3) Char → for character & string

E.g. 'A' "Hello", "Hello 123"

(4) double → For double precision floating value.

E.g. double x;

— Input Function & Output Function —

(1.)-Input function →

Scanf() → To input the value of a variable through Keyboard.

Syntax :- Scanf("control string", variable — Address);

Syntax: Scanf("control string", Variable-Address);

E.g. Int l, b

Suppose we are going to Input the value of l and b.

(i) Scanf("%d", &l);

(ii) Scanf("%d", &b)

(iii) Scanf("%d %d", &l, &b)

Now we can use the keyboard to Input a value Suppose we Input 50, so, the value of l will become 50.

%d → Int
%f → float
%c → Char
%lf → double

& → at the adress of or Ampersand

(2) Output function →

Printf() → This function is used to print the result on the output screen. It is also used to print the message on the output screen.

Syntax: Printf("control string", variable_name);

E.g. Suppose,

Int A = l * b

result is stored in A. and we have to print the value of A.

(i) Printf ("%d", A);
(ii) Printf ("Area = %d", A);
(iii) Printf ("%d %f", P, Q);
(iv) Printf ("Area of Rectangle : %d", A);
(v) Printf ("I Live in Lucknow");

Suppose value of A is 60

Area = 60

Area of Rectangle : 60

I Live in Lucknow.

Que. Write a program to print "Hello India".

```
# include <Stdio.h>
# include <conio.h>
Void main ( )
{
   Printf ("Hello India");
}
```

Stdio → Standard Input / output
Scanf() Printf()

→ Preprocessor directive.
.h → header file
Conio → Console Input / output
e.g. getch(), clrscr()
Main() → Program Execution (Translate into binary) starts from main() function Every program must have only one main() function.

Que- write a program to find the area of rectangle.

```
# Include <stdio.h>
# Include <conio.h>
Void main ( )
{ int l, b, a ;
Printf("Input l and b");
Scanf ("%d %d", &l &b);
        a = l * b;
Printf ("Area = %d", a)
        getch ( ); }
```

Run 1 (output)

Input l and b
20
15
Area = 300

Ans Write program to find S.I.

```
#include <stdio.h>
#include <conio.h>
 void main ( )
{   float  P, n, t , SI ;
    Printf  ("Enter P, n, t") ;
    Scanf ("%f %f %f", &P, &n, &t);

      SI = P * n * t / 100
   Printf ("Simple Interest = %f", SI);
      getch ( );
                }
```

Que. write a program to find area and circumference of circle.

```
#include <stdio.h>
#include <conio.h>
  void main ( )
  { float  r, a, c;
    Printf ("Enter radius");
    Scanf ("%f, &r);
    a =  3.14 * r * r;
     c =  2 * 3.14 * r;
   Printf ("Area = %f", a);
   Printf (" circumference = %f", c);
          getch( );
  }
```

Conditional control statements :—

(1) If statement – It is one way decision statement, in this if result true then the statement or statement followed by if will execute.

(2) If-else statement – It is two way decision statement, In this if cond^n is True, the

(3) If-else-if ladder – it is used to test the set of condition in sequence.

(4) nested if – when one condition depend on another condition and so on. then we use

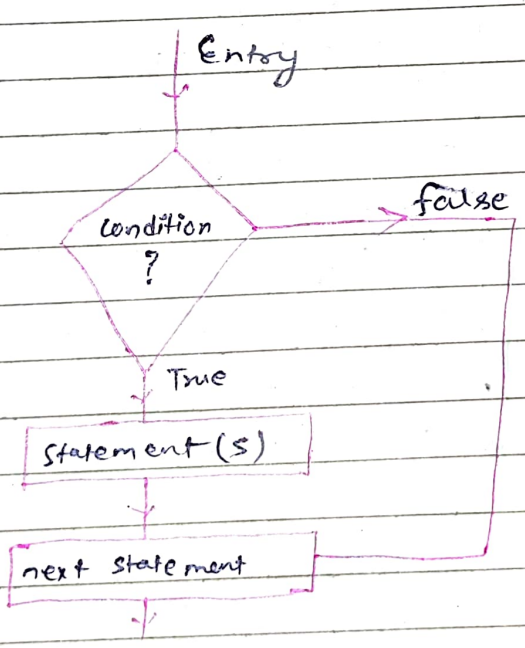
(5) Switch case – when we want to solve multiple option type problem but we need to choose one at a time then we use switch case.

(1) if statement :->

Syntax :

```
if condition
{ statement(s);
}
```

It is a one way direction statement, in this if the result of the condition is True, then the statement or statement followed by if will Execute.



Q) Write a program to check whether the numbe is negative

Sol)

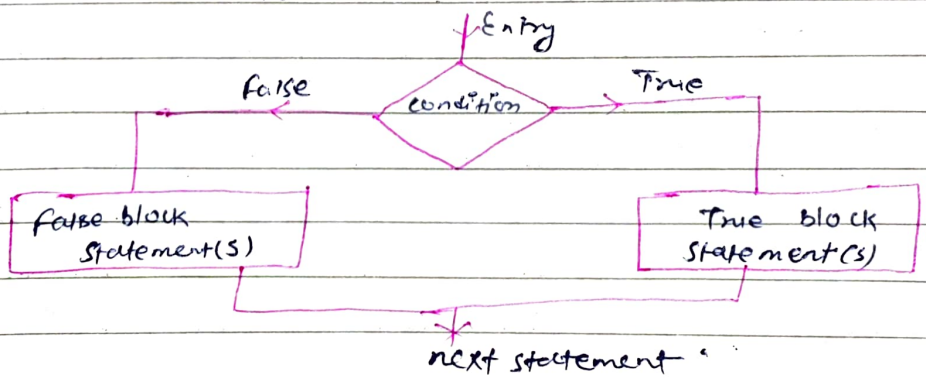
```
#include <stdio.h>
#include <conio.h>
void main()
{   int x;
printf("Enter a number");
scanf("%d", &x);
if (x<0)
{ printf("%d is negative", x);
printf("Hello");
    getch();
}
```

2) if-else statement →

Syntax:

```
if (condition)
{ True block statement(s);
  }
else
   { false block statement(s);
     } next statement;
```

If is a two way decision statement, in this if the result of the condition is True, then the true block statement(s) will Execute otherwise false block statement (s) will Execute



2 [illegible]

Write a program to find greater between two numbers -

```
# include <stdio.h>
# include <conio.h>
 Void main ( )
{ int  x, y;
 Printf ( "Enter two number x, y");
Scanf ("%d %d", &x, &y );
if ( x > y)
{ printf ("%d", &x is greater); Printf ("%d is greater", x);
}
 else
   { printf ("%d is greater", y);
    }
      getch ( );
        }
```

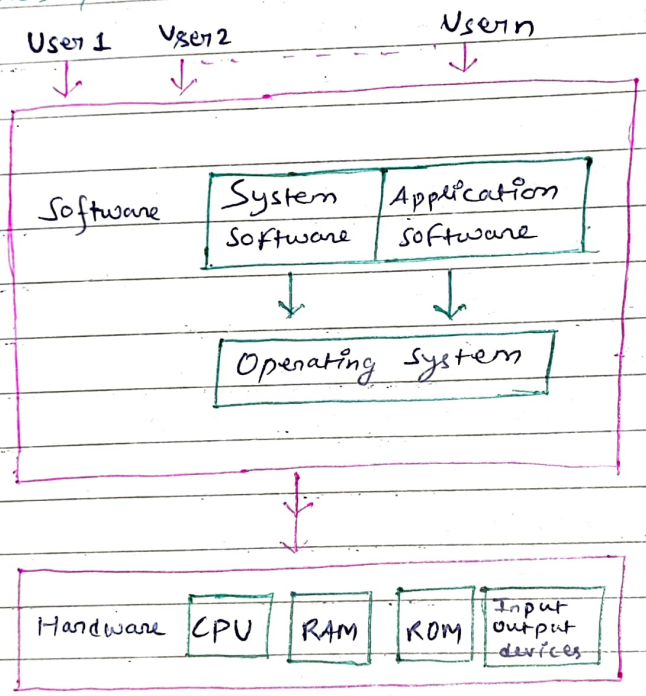
write a program to find whether a given number is even or odd.

```
# include <stdio.h>
# include <conio.h>
Void main ( )
{    Int x;
Printf ( "Enter a number");
Scanf ("%d", &x);
if (x%2 ==0)
{ printf ("%d is even", x);
  }
  else
    { Printf ("%d is odd ", x);
      }
      getch ( );
        }
```

Operating System :- Operating System is a program which acts as an interface between the user of the computer and computer hardware.

" It is a primary system software which is used to manage and control hardware and software resources of computer system " - Some popular operating system are -

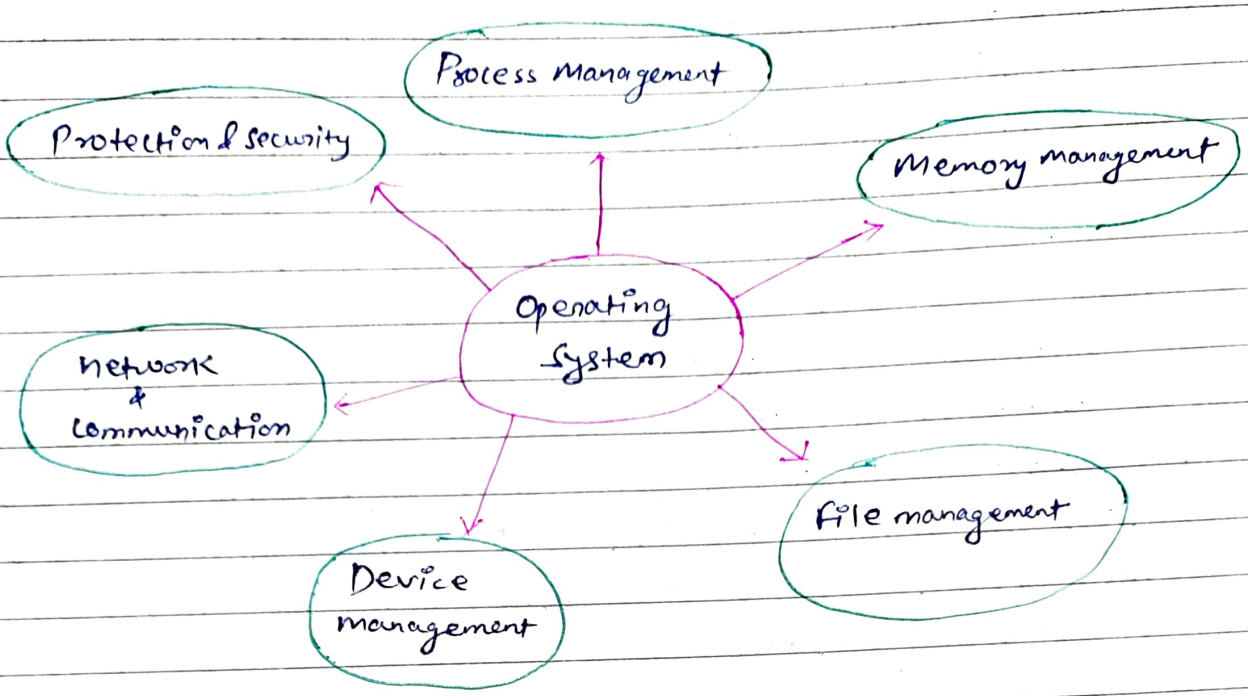
Windows, Linux, Unix, Android etc.



Structure of operating system.

Function / Goal / Purpose of operating system

Various function performed by operating system are as follows:-



Process Management :— Execution of a process requires various computer resources such as—
CPU time, RAM, Input output devices, process management Deals with computing and distributing various resources to different processes it is done by the operating system.

Memory management :— Operating system is Responsible for allocation and deallocation of memory (RAM). to different processes.
it decides which process is to be load. in memory.

File management :— Permanent storage of data is an essential requirement of any computer system, for this data is to Return secondary storage devices in form of file. File management done by Operating system.

Device management :— A computer system have various input and output devices such as keyboard, mouse, monitor, operating system allocate and deallocate devices to different processes.

Networking & communication :— Operating system helps in networking and communication.

Protection & security :— Operating system helps to protect and secure our computer system.

Types of Error

A Mistake in program is called an Error.

— There are ~~three~~ three types of Error — Bug.

(1) ~~Bugs~~ Compile type Error - the Error generated at compile time is called compile type error. it can be of two types —

1. Syntax Error → when Rule of C-programming language is not Followed then the Error generated is called syntax Error.

Ex— int a,b // in this statement semi colon is not present so error generated statement missing.

2. Semantic Error → This type of Error is generated when statement is not meaningful to the compiler.

Ex → x + y = z; // in this statement Error will Generate l-value required.

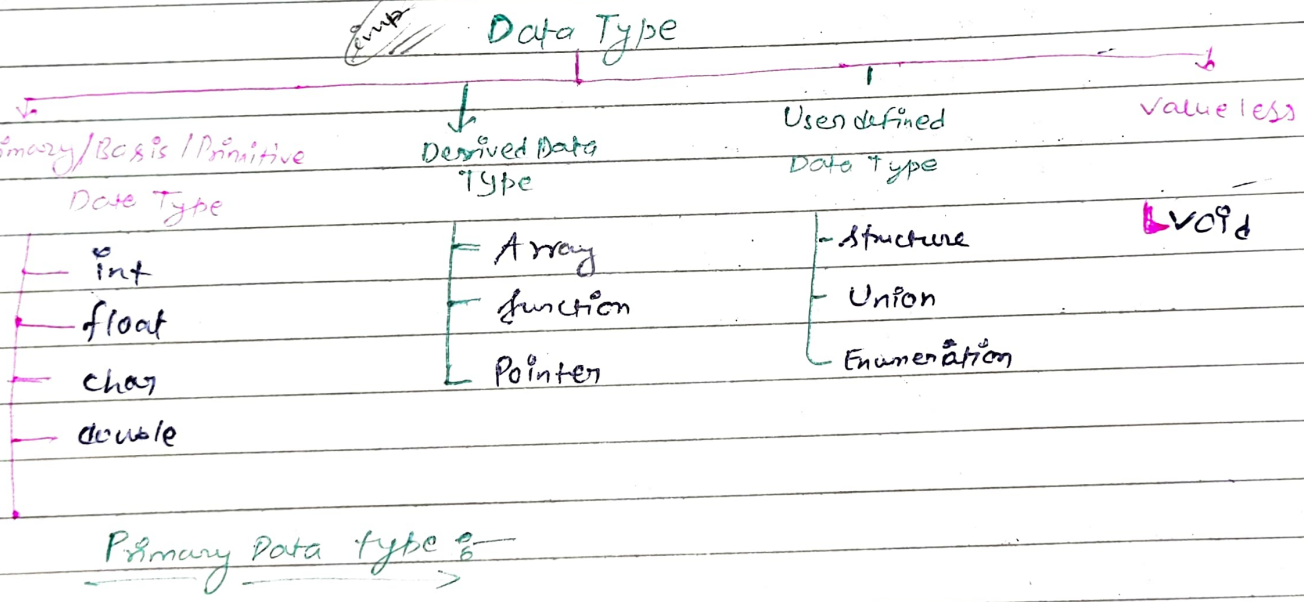
(2) Logical Error → This type of Error is generated by the logical mistake of the programmer. It is not shown by the compiler.

Eg → Sum = a - b; // In this use are doing sum but we have written (-) sign in place of (+).

(3) Run time Error → Error that occure during Execution of program is Run time Error.

Eg → Domain Error, memory full.

Data Types → Data Types indicates type of the type of data that a variable can hold in 'c' language data types can be divided into the following type



Primary Data type →

(1) int → used for Integers (numbers without decimal)

Eg: 5, -10, 100 etc.

Syntax: | Data Type | Variable name; |

Eg: int x; // x is variable and the value of x would be an integer value.

(2) float → Used for numbers with decimal.

E.g: 5.35, 0.27, -7.36 etc.

E.g: float x; // x is variable and the value of x would be a float value.

(3) char → Used for character and string

E.g: 'A', 'x', "Hello123" etc.

Eg: char x;

(4) double → Used for double precision floating value

E.g: double x;

Data type Modifier →

Data type modifier modify the range & Basic Arithmatic properties of base.

- Signed
- Unsigned
- Short
- long

Size, Range & Format specifier of Basic data type (16 bit machine).

Data Type	Size (in Byte)	Range	Format specifier.
char or signed char	1 Byte	-128 to 127	%c
unsigned char	1 Byte	0 to 225	%c
signed int, unsigned int, short int or signed short int, unsigned short int.	2 Byte	-32768 to 32767	%d
	2 Byte	0 to 65535	%u
	1 Byte	-128 to 127	%d
	1 Byte	0 to 255	%u
long int or signed long int	4 Byte	-2147483648 to 2147483647	%ld
unsigned long int	4 Byte	0 to 4294967295	%lu
float	4 Byte	3.4E-38 to 3.4E+38	%f
double	8 Byte	1.7E-308 to 1.7E+308	%lf
long double.	10 Byte	3.4E-4932 to 1.1E+4932	%Lf

% → Modulus Operator
(use to find the Remainder)

== | Equality operator
(Comparison whether both are Equal or not)

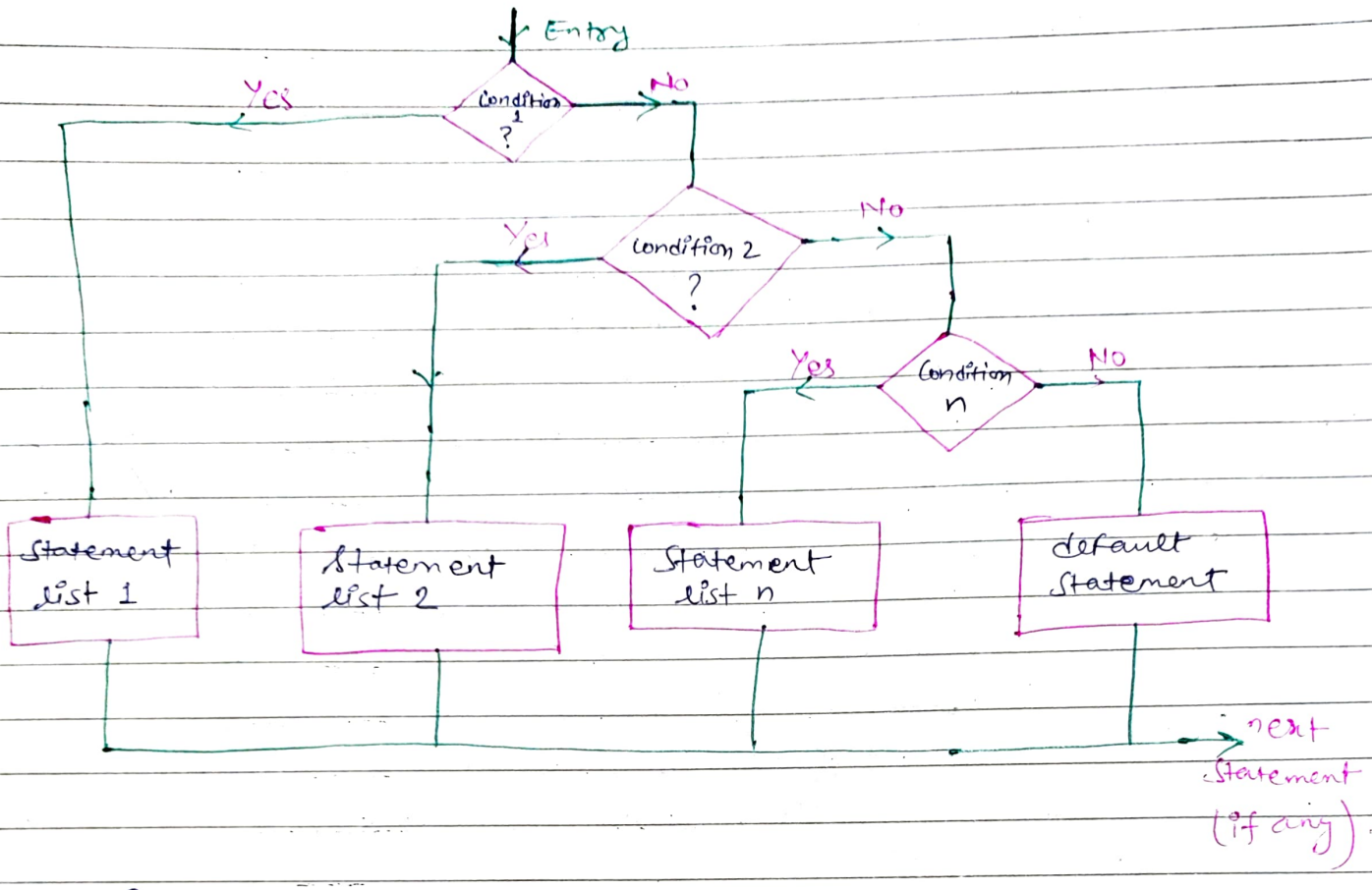
Eg. x=5 | the value of x is 5

Eg. x==5 | We are checking whether the value of x is Equal to 5 or not

if-else-if ladder

Syntax:—

```
if (condition 1)
    Statement list 1;
else
    if (condition 2)
    Statement list 2;
    - - - - - - - - -
else
    if (condition n)
    Statement list n ;
else
    default statement;
```



If is used to test set of conditions in sequence. when a condition becomes True then statement associated with that condition will Execute and rest of the condition will by pass. If none of the condition becomes True then default statement will Execute.

Que- write a program to check whether a number is positive, negative or zero:

Sol⇒

```
#include <stdio.h>
#include<conio.h>
Void main()
{
  Int x; Enter two number x, y");
Printf("Enter a number");
Scanf("%d", &x);
if (x>0)
Printf("%d is positive", x);
else
  if (x<0)
  Printf("%d is negative", x);
else
  Printf("Number is Zero");
    getch();
}
```

RUN
Enter a no.
15 is positive

Que write a program to input the selling price and cost price of a product and calculate profit, loss or no profit, no loss.

Sol⇒

```
#include<stdio.h>
#include<conio.h>
#include<math.h>
  Void main()
{
  Int SP, CP, P, L;
Printf("Enter selling price & cost price");
Scanf("%d %d", &SP &CP);
```

```
if (SP>CP)
{ P= CP- SP
Printf ("profit = %d",P);
}
else
    if (CP>SP)
{   l = CP-SP ;
Printf ("loss = %d",l);
}
 else
    Printf ("No profit no loss");
      getch ( );
}
```

Q, Write a program to print the division of a student as given by the condition.

Percentage	division
>= 60	first division.
above or equal to 50 and less than 60	second division
above or equal to 40 and less than 50	Third division
< 40	fail.

```
#include <stdio.h>
#include <conio.h>
 void main( )
{ float Percentage;
  Printf ("Enter Percentage");
  Scanf ("%f", & Percentage);
  if (Percentage >= 60)
```

```
Printf ("First division");
else
   if (Per >= 50)
Printf (" Second division");
else
 if ( Per >= 40)
 Printf (" Third division");
 else
    Prinf (" Fail");
      getch ( );
    }
```

Logical AND (&&) :—

A	B	A && B
False	false	False
false	False	False
~~False~~ True	~~False~~ False	False
True	True	True

```
## include <Stdio.h>
# include <conio.h>
 Void main ( )
{ float  Percentage ;
 Print f ("Enter Percentage");
 Scanf ("%d", & Percentage);
 if ( Per >= 60)
 Printf (" First division");
 else
    if (Per >= 50 && per < 60)
    Printf ("Second division");
else
     if (Per >= 40 && Per < 50)
     Printf ( Third division);
       else
          printf ("Fail"); getch( ); }
```

Que- write a program to check whether a number is positive, negative or zero:

Sol^n =>

```
# include <stdio.h>
# include<conio.h>
Void main ( )
{
    int x; ~~Enter two numbers x, y"~~);
Printf ("Enter a number");
Scanf ( " %d", & x);
if  (x>0)
Printf (" %d is positive ", x);
else
  if ("x<0)
  Printf ("%d is negative ", x);
else
   Printf (" Number is Zero ");
      getch ( );
   }.
```

RUN
Enter a no.
15 is positive

Que write a program to input the selling price and cost price and of a product and calculate profit, loss or no profit, no loss.

Sol^n =>

```
# include< stdio.h>
# include< conio.h>
~~# include <math.h>~~ x
  void main ( )
{
  int SP, CP , P, l ;
Printf (" Enter selling price & cost price");
Scanf ( "%d %d", & SP & CP);
```

```
Pf ( SP>CP)
{ P= CP- SP
Printf ("profit = %d",P);
}
else
    PF ( CP>SP)
    {  l = CP-SP ;
    Printf ("loss = %d",l);
    }
    else
        Printf ("No profit no loss");
        getch ( );
    }
```

Ans, Write a program to print the division of a student as given by the condition.

Percentage	division
>=60	First division.
above or equal to 50 and less than 60	Second division
Above or equal to 40 and less than 50	Third division
<40	fail.

```
#Include<stdio.h>
#Include<conio.h>
  Void main( )
{  float Percentage;
   Printf ("Enter Percentage");
   Scanf ("%f", & Percentage);
   Pf (Percentage >=60)
```

```
Printf ("First division");
else
   if (Per >= 50)
Printf ("Second division");
else
  if ( Per >= 40)
Printf ("Third division");
else
   Printf ("Fail");
      getch ();
     }
```

Logical AND (&&) :—

A	B	A && B
False	False	False
False	False	False
~~True~~	~~False~~	False
True	True	True

```
#include <stdio.h>
#include <conio.h>
  void main ()
{ float Percentage;
Printf ("Enter Percentage");
Scanf ("%d", & Percentage);
if ( Per >= 60)
Printf ("First division");
else
   if (Per >= 50 && per < 60)
   Printf ("Second division");
else
     if (Per >= 40 && Per < 50)
     Printf ("Third division");
       else printf ("Fail"); getch() }
```

Que- Write a program to find greatest among three numbers.

```
#include<stdio.h>
#include<conio.h>
void main()
{
int x,y,z;
Printf("Enter three number");
Scanf("%d %d %d",&x,&y,&z);
if(x>y && x>z)
Printf("Greatest=%d",x);
else
if(y>x && y>z)
Printf("Greatest=%d",y);
else
Printf("Greatest=%d",z);
getch();
}
```

Que A company decides to give Bonus to all its employee on diwali. a five percent Bonus on salary is given to male workers and Ten percent Bonus to female workers on Salary. Write a program to Input Salary and gender of employe, if salary of employee is less than ten thousand then employee get an extra Two percent Bonus on salary. calculate Bonus and salary that has to be given to the employee.

```
#include <stdio.h>
#include <conio.h>
  void main( )
{  Char gender;
   Float salary, bonus;
Printf ("Enter gender 'm' for male & 'F', for Female");
Scanf ("%c", & gender);
 Printf (" Enter your salary");
 Scanf ("%f", & salary);
If (gender == 'm')
 bonus = salary * 0.05;
 else
 bonus = salary * 0.10;
   If (salary < 10000)
 bonus = bonus + (salary * 0.02);
  Salary = salary + bonus;
Printf (" Final salary = %f," Salary );
Printf ("Total bonus = %f", bonus);
     getch ( );
}
```