

**WEL - COME**  
**CLASS 11**  
**COMPUTER SCIENCE**

# Important C programming question

- 1) What is control cont. statement? Wap to select and prints largest number among three
- 2) What is operator? Explain different types of operators used in c programming with example
- 3) Wap to display odd, even and zero number.
- 4) Wap to display positive , negative and zero number.
- 5) Wap to display number which divisible by 7, 5 or not
- 6) Describe the switch statement with example
- 7) wap which find the sum , different, and product of 2 number using switch case statement.
- 8) What is statement ? Describe the types statement.
- 9) What is keywords? Define data type in c programing.
- 10) Define input / output function in c programming.
- 11) What is expression? Define variable and constants.

# Welcome to C Programming

## Introduction to C



# INTRODUCTION :

**C is a general-purpose programming language that is extremely popular, simple and flexible. It is machine-independent, structured programming language which is used extensively in various applications.**

**C was the basic language to write everything from operating systems (Windows and many others) to complex programs like the Oracle database, Python interpreter and more.**

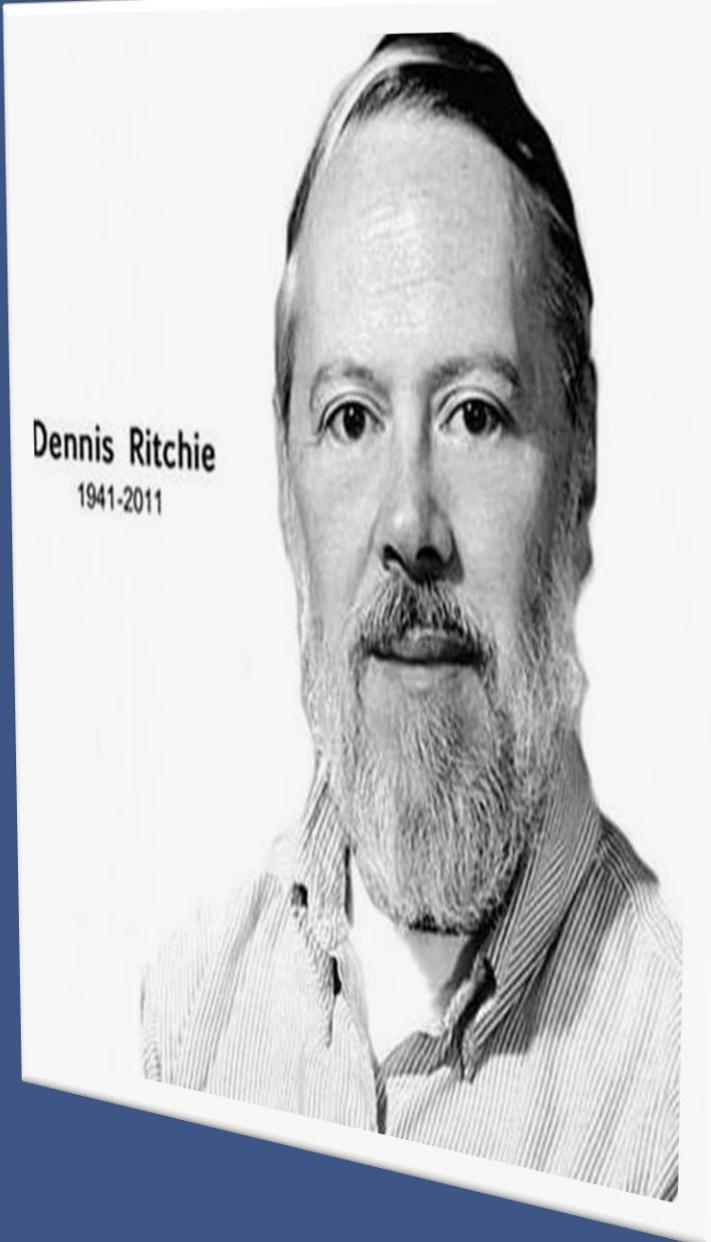


Dennis Ritchie  
1941-2011

# HISTORY OF C PROGRAM

A high-level programming language developed by Dennis Ritchie at Bell Labs in the mid 1970s. Although originally designed as a systems programming language, C has proved to be a powerful and flexible language that can be used for a variety of applications, from business programs to engineering. C is a particularly popular language for personal computer programmers because it is relatively small -- it requires less memory than other languages.

The first major program written in C was the UNIX operating system, and for many years C was considered to be inextricably linked with UNIX. Now, however, C is an important language independent of UNIX.



## **Advantages of C Language**

- a. C language is a building block for many other currently known languages. C language has variety of data types and powerful operators. Due to this, programs written in C language are efficient, fast and easy to understand.**
- b. C is highly portable language. This means that C programs written for one computer can easily run on another computer without any change or by doing a little change.**
- c. There are only 32 keywords in ANSIC and its strength lies in its built-in functions.**
- c. C language is a structured programming language. This makes user to think of a problem in terms of function modules or blocks. Collection of these modules makes a complete program. This modular structure makes program debugging, testing and maintenance easier.**
- f. C is portable language.**
- g. C language is easy for beginners.**
- h. C language support system programming.**
- i. C language support no of operators.**
- j. C is the collection of lot of library files.**

# **Disadvantages of C Language**

- a. C does not have concept of OOPs, that's why C++ is developed.**
- b. There is no runtime checking in C language.**
- c. C doesn't have the concept of namespace.**
- d. C doesn't have the concept of constructor or destructor.**
- e. C language has no script type checking.**
- f. C language has no run time checking mechanism.**
- g. C does not support oops features.**

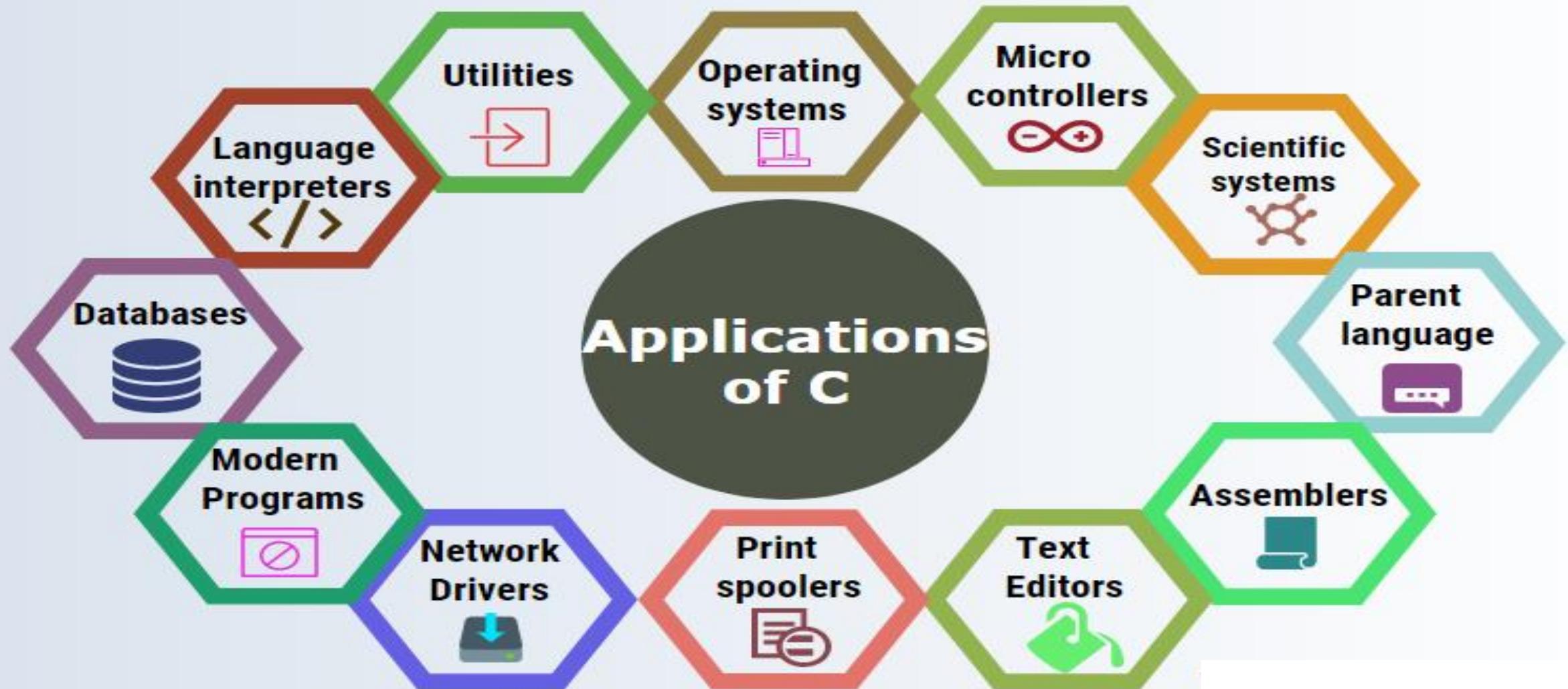
# Where is C used? Key Applications

- ❖ 'C' language is widely used in embedded systems.
- ❖ It is used for developing system and applications.
- ❖ It is widely used for developing desktop applications.
- ❖ Most of the applications by Adobe are developed using 'C' programming language.
- ❖ It is used for developing browsers and their extensions. Google's Chromium is built using 'C' programming language.
- ❖ It is used to develop databases. MySQL is the most popular database software which is built using 'C'.
- ❖ It is used in developing an operating system. Operating systems such as Apple's OS X, Microsoft's Windows, and Symbian are developed using 'C' language. It is used for developing desktop as well as mobile phone's operating system.
- ❖ It is used for compiler production.
- ❖ It is widely used in IOT applications.

# Why learn c?

- 'C' is a base language for many programming languages.
- learning 'C' as the main language will play an important role while studying other programming languages.
- It shares the same concepts such as data types, operators, control statements and many more.
- 'C' can be used widely in various applications.
- It is a simple language and provides faster execution.

# Application area of C



# Install in C

## TURBO C

## TURBO C++

## TC Editor

TC Editor is very simple and easy to use; here i will give you all tips related to TC Editor and some shortcut keys related to TC Editor which is very useful at the time of coding. Turbo C is a most common C language compiler. Below i will discuss all about its Interfaces.



- File:** This menu contains group of commands used for save , edit , print program, exit from Turbo C editor etc.
- Edit:** This menu contains group of commands used for editing C program source code. Example Copy, Cut, Paste, Undo etc.
- Search:** This menu contains group of commands used for searching specific word as well as replacing it with another one.
- Run:** This menu contains group of commands used for running C program.
- Compile:** This menu contains group of commands used for compiling C program.
- Debug:** This menu contains group of commands used for debugging C program.
- Project:** This menu contains group of commands used for opening, closing and creating projects.
- Options:** This menu contains group of commands used for configuring IDE of Turbo C and setting up directories etc.
- Windows:** This menu contains group of commands used for opening, closing various windows of IDE.
- Help:** This menu is used to get help about specific topic of C language. Similarly to get help about a specific keyword or identifier of C.

# How 'C' Works?

- **c is a compiled language.**
- **A compiler is a special tool that compiles the program and converts it into the object file which is machine readable.**
- **After the compilation process, the linker will combine different object files and creates a single executable file to run the program.**



# Summary

- **'C' was developed by Dennis Ritchie in 1972.**
- **It is a low programming level language close to machine language**
- **It is widely used in the software development field.**
- **It is a procedure and structure oriented language.**
- **It has the full support of various operating systems and hardware platforms.**
- **Many compilers are available for executing programs written in 'C'.**
- **A compiler compiles the source file and generates an object file.**
- **It is highly portable.**

# Header files

**Header files contain definitions of functions and variables, which is imported or used any c program by using the per- processor # include statement.**

#include <file-name>.h

Example:

```
#include<stdio.h>           //Pre-processor directive
#include<conio.h>            //pre- processor directive
void main()                  //main function declaration
{
    Clrscr();
    printf("Hello World");    //to output the string on a display
    getch();                 //terminating function
}
```

# **Shortcut keys Related to TC Editor**

- Alt + x : Close TC Editor.**
- Ctrl + f9 : Run C Program.**
- Alt + f9 : Compile C Code.**
- Alt + Enter : Get Full Screen or Half Screen TC Editor.**
- Ctrl + y : Delete complete line above the cursor.**
- Shift + Right arrow : Select Line of Code.**
- Clt + Insert : Copy.**
- Shift + Insert : Paste.**
- Shift + Delete : Delete.**

# Structure of C program

- **Preprocessor Commands:** These commands tell the compiler to do preprocessing before doing actual compilation. Like `#include <stdio.h>` is a preprocessor command which tells a C compiler to include stdio.h file before going to actual compilation.

**Function:** Function is a code segment or module which helps you to perform a task quick and easily.

- **Variable:**

A Variable is a symbolic unique name that occupies space in the computer memory to store data temporarily during run time.

- **Statements & Expressions:**

An expression is the valid combination of numeric or string constants and variables with operators which returns a single value.

**Comments:** A comment is an explanation or description of the source code of the c program. It helps a developer explain logic of the code and improves program readability.

## **Fundamentals of C:**

### **Character set:**

**The character set is the fundamental raw material of any language they are used to represent information.**

**A character set in 'C' is divided into:**

- Letters**
- Numbers**
- Special characters**
- White spaces (blank spaces)**

## **What Is Comment In C Language?**

**A comment is an explanation or description of the source code of the c program. It helps a developer explain logic of the code and improves program readability.**

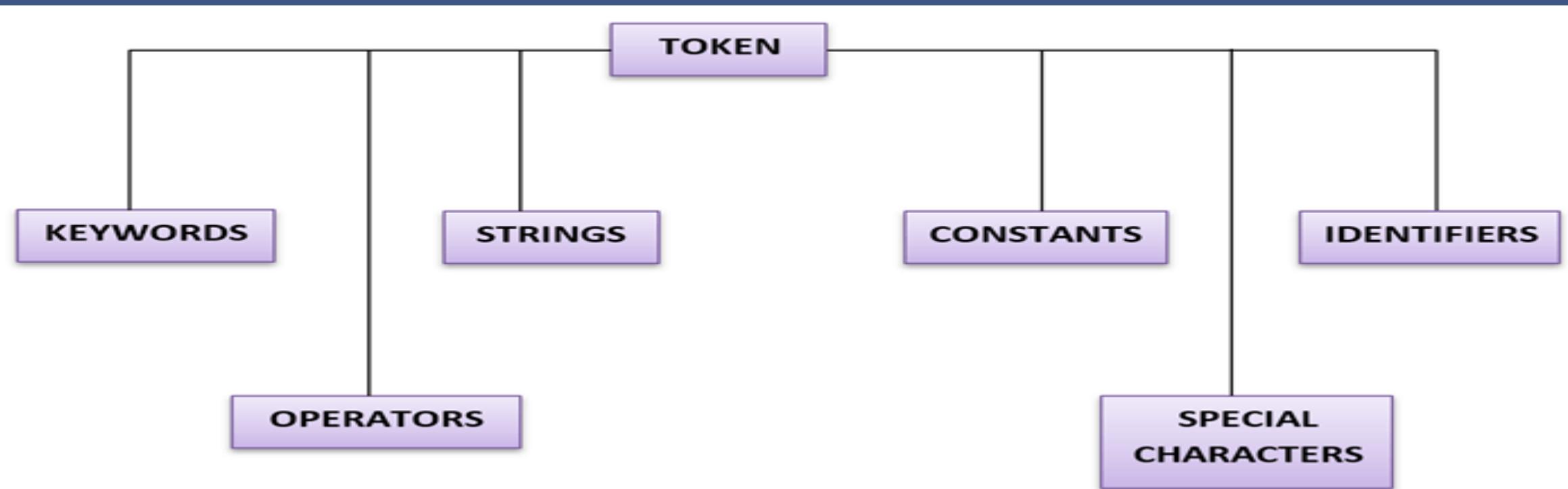
**A comment that starts with a slash asterisk /\* and finishes with an asterisk slash \*/ and you can place it anywhere in your code, on the same line or several lines.**

**2) Single-line Comments which uses a double slash // dedicated to comment single lines**

# C token

C tokens are objects by which you build a c program.

Each and every smallest individual units in a c program are known as c tokens.



# Keywords

Keywords have fixed meanings, and the meaning cannot be changed. There are a total of 32 keywords in 'C'. Keywords are written in lowercase letters.

Following table represents the keywords in 'C'-

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

# Operator:-

**Operator is a special symbol that tells the compiler to perform specific mathematical or logical Operation. Operator means to operate the operands.**

**Example:       $p = a+b$  ; Here, '=' & '+' are operators, 'a' & 'b' are operand and ' $p = a+b$ ' is an operation.**

**The data item act upon operator are called operands. The portion which indicates the action to be performed on operands is called operators.**

There are different types of operators:

1. Arithmetic operators:
2. Relational operators
3. Equality operator
4. Logical operators
5. Assignment operators
6. Unary operators
7. Conditional operators
8. Size of operator
9. Comma operator

## Data types:

**Data types is the characteristic or set of possible values that can be stored in a variable.**

### Data Types in C

#### Primary data types

- ◆ 01\_Int
- ◆ 02\_Float
- ◆ 03\_Cha
- ◆ 04\_Void

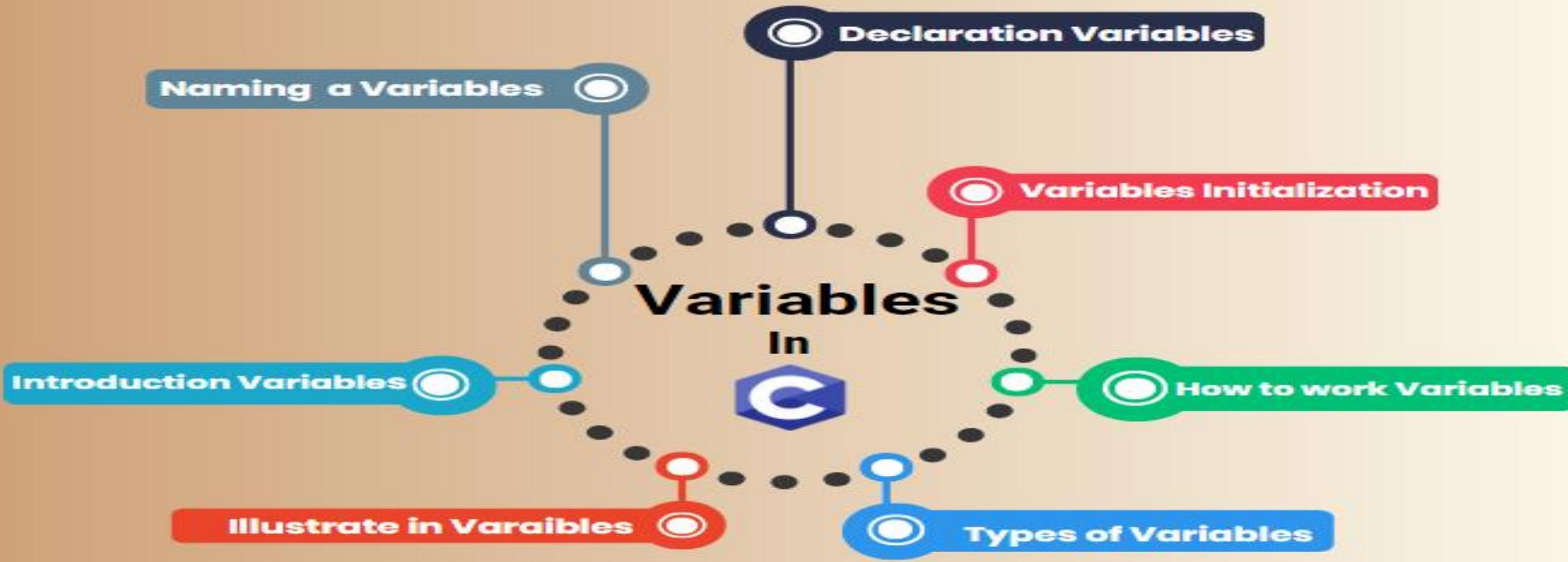
#### Derived data types

- ◆ 01\_Array
- ◆ 02\_Structure
- ◆ 03\_Union
- ◆ 04\_Pointer



# Variable:

A Variable is a symbolic unique name that occupies space in the computer memory to store data temporarily during run time.



## **Types of variable:**

- ✓ **Local variable:**
- ✓ **Global Variable:**
- ✓ **Static Variable:**

## **Local Variable:**

**Variables that are declared inside a function or block are called local variables. They can be used only by statements that are inside that function or block of code.**

**Following is the example using local variables. Here all the variables a, b and c are local to main() function.**

```
#include <stdio.h>
#include<conio.h>
Void main ()
{
    /* local variable declaration */
    int a, b;
    int c;
    /* actual initialization */
    a = 10;    b = 20;  c = a + b;
    printf ("value of a = %d, b = %d and c = %d\n", a, b, c);
    getch();
}
```

## **Types of variable:**

- ✓ **Local variable:**
- ✓ **Global Variable:**
- ✓ **Static Variable:**

## **Global variable:**

**Global variables are defined outside of a function, usually on top of the program.**

**The global variables will hold their value throughout the lifetime of your program and they can be accessed inside any of the functions defined for the program.**

```
#include <stdio.h>
#include<conio.h>
/* global variable declaration */
int g;
Void main ()
{
/* local variable declaration */
int a, b;
/* actual initialization */
a = 10;
b = 20;
g = a + b;
printf ("value of a = %d, b = %d and g = %d\n", a, b, g);
getch();
}
```

## Types of variable:

- ✓ **Local variable:**
- ✓ **Global Variable:**
- ✓ **Static Variable:**

## Static variable:

A static variable is a variable that has been allocated statically—whose lifetime or "extent" extends across the entire run of the program. :

```
#include <stdio.h>
/* function declaration */
void func(void);
static int count = 5;
/* global variable */
main() {
    while(count--)
    {
        func();
        getch();
    }
    /* function definition */
    void func( void )
    {
        static int i = 5;
        /* local static variable */
        i++;
        printf("i is %d and count is %d\n", i, count);
    }
}
```

# Constant:

Constants are the fixed values that never change during the execution of a program

a) integer constants: E.g. 54312

7777

b) floating point constants:

c) character constants

Example: 'B', '2', '\$', 'f'

d) string constant:

E.g. "Green", "Washington", "D.C. 2005", "2\*(l+3)/j", " "

E.g.	
Valid	Invalid
7.0	7
17.6	17.
-14.5	-14.5.456
0.8	.8

# Constant:

Constants are the fixed values that never change during the execution of a program

a) integer constants: E.g. 54312

7777

b) floating point constants:

c) character constants

Example: 'B', '2', '\$', 'f'

d) string constant:

E.g. "Green", "Washington", "D.C. 2005", "2\*(l+3)/j", " "

E.g.	
Valid	Invalid
7.0	7
17.6	17.
-14.5	-14.5.456
0.8	.8

# **Introduction to Library Function: Input / Output (I/O) Functions**

# **Statements :-**

**A statement is a command given to the computer that instructs the computer to take a specific action, such as display to the screen, or collect input.**

**Simple statement:**

Simple statements are those statements which are ended with semicolon and ready to perform some action.

**Example:**

`a=33;`

`c=a+b;`

`printf("Sum=%d",sum);`

## **Compound statement:**

**A compound statement consists of several individual statements enclosed within a pair brackets {}.**

Example:

```
{  
pie=3.141593;  
circumference=2*pie*radius;  
area = pie*radius*radius;  
}
```

## **Control statement:**

**Combination of both statement which is used to create special program feature such as logical tests loops and branches.**

```
main()  
{  
int n=5,count=1;  
int sum;  
while (count<=n)  
{  
printf("n=");  
sum=sum+n;  
count++;  
}  
Printf("%d",sum);  
}
```

# Control Structure:

A statement that is used to control the flow of execution in a program is called control structure. It combines instruction into logical unit.

## Types of control structures

1. Sequence
2. Selection
3. Repetition/Looping

### Sequence:

Sequence is the process which solved required problem with out conduction and logic.

### Syntax:

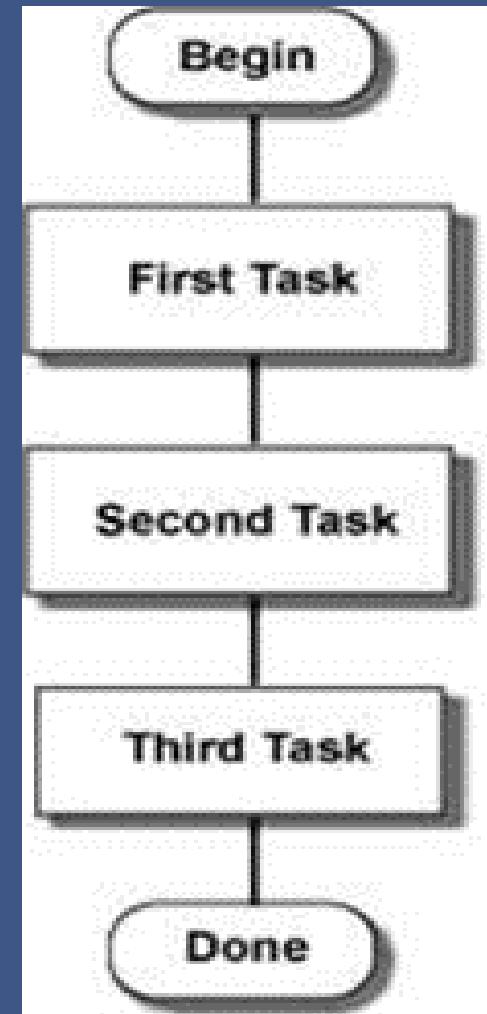
Start Instruction 1;

Instruction 2;

Instruction 3;

....

.... Instruction n; End

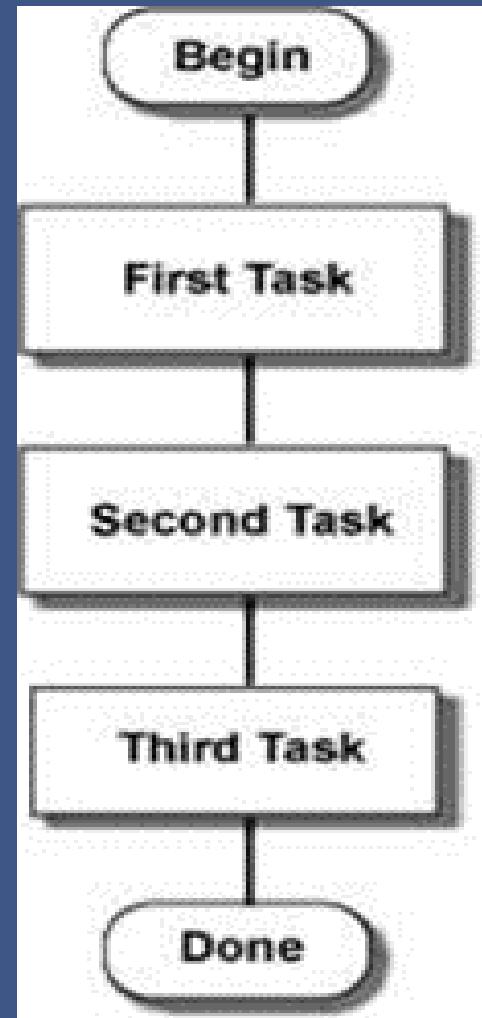


# Control Structure:

Example:

A C program to read two numbers and find sum of their square:

```
#include<stdio.h>
#include<conio.h>
void main()
{
clrscr();
int a,b;
int sum=0;
printf("enter two numbers");
scanf("%d%d",&a,&b);
sum=a*a+b*b;
printf("\nThe sum of squares of two nos. = %d",sum);
getch();
}
```



## **Some important question related to Sequence**

- a. WAP to display area of circle with flowchart.
- b. WAP to display simple interest.
- c. WAP to display area and perimeter of rectangle.

### **Selection:**

**Selection is the process of solving required problem with conduction and logic. Types of selection:**

- a. If statement
- b. If else statement
- c. If else if statement
- d. Switch case statement.

## Some important question related to Sequence

- a. WAP to display area of circle with flowchart.
- b. WAP to display simple interest.
- c. WAP to display area and perimeter of rectangle.

**condition**

**If( $n > 0$ ) = positive**

**If( $n < 0$ ) = negative**

**if( $n \% 2 == 1$ ) odd number**

**If( $n \% 2 == 0$ ) even number**

**If( $n \% 7 == 0$ ) divisible by 7**

**If( $a > b$ ) a is largest number among two**

**If( $a > b \ \&\& \ a > c$ ) a is largest number**

**If statement:** It is one of the powerful conditional statement. If statement is responsible for modifying the flow of execution of a program. If statement is always used with a condition. The condition is evaluated first before executing any statement inside the body of If.

Syntax:

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

**Example:**

**WAP to display positive number with flow chat**

```
#include<stdio.h>
#include<conio.h>
void main()
{
    clrscr();
    int N;
    printf("enter the required value N");
    Scanf("%d",&N);
    If(N>0)
    {
        Printf("N is positive Number");
    }
    getch();
}
```

### **IF related Questions:**

- 1. WAP to display negative number.**
- 2.WAP to display odd number.**
- 3.WAP to display even number.**
- 4.WAP to display number which divisible by 7.**

# If ..... Else statement:

In if-else statement if the condition is true, then the true statement(s), immediately following the if-statement are executed otherwise the false statement(s) are executed.

Syntax:

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

## Example:

**WAP to display positive or negative number**

**with flow chat**

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

```
void main()
```

```
{
```

```
clrscr();
```

```
int N;
```

```
printf("enter the required value N");
```

```
Scanf("%d",&N);
```

```
if(N>0)
```

```
{
```

```
Printf("N is positive Number");
```

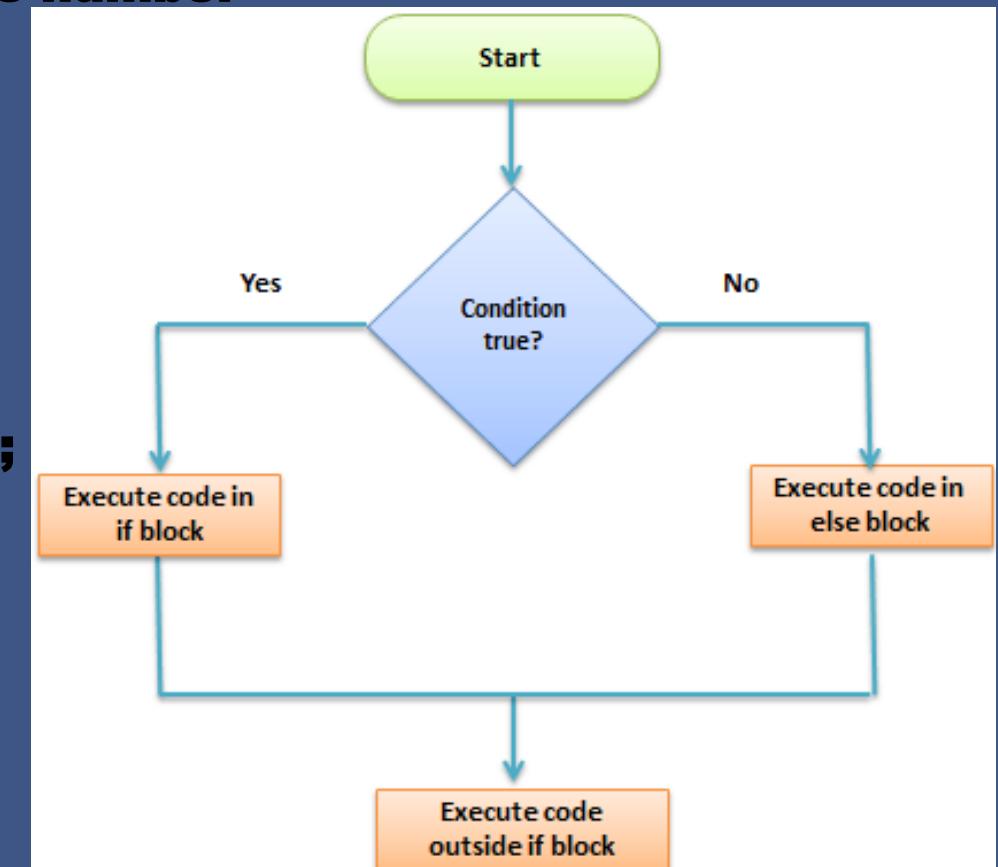
```
else
```

```
Printf(N is negative number");
```

```
}
```

```
getch();
```

```
}
```



## **If ..... else related question**

- 1. WAP to display odd or even number with flow chat.**
- 2. WAP to display positive or negative number with flow chat.**
- 3. WAP to display number which divisible by 7 or not.**
- 4. WAP to display largest number among two number.**

### **iii) if ....else if statement**

**It can be used to choose one block of statements from many blocks of statements. The condition which is true only its block of statements is executed and remaining are skipped.**

**Example:**

**WAP to display positive ,negative and zero number with flow chat.**

```
#include<stdio.h>
#include<conio.h>
void main()
{
    clrscr();
    int N;
    printf("enter the required value N");
    Scanf("%d",&N);
    if(N>0)
    {
        Printf("N is positive Number");
    else if( N<0)
        Printf(N is negative number");
    else
        Printf("n is zero number");
    }
    getch();
}
```

## If.....else if related questions:

- a. WAP to display odd even and zero number with flow chat.
- b. WAP to display smallest number among three number with flow chat.
- c. WAP to display largest number among four number.
- d. WAP to test whether the entered character is alpha digit or alpha character or special character on the basis of given condition.

Character	Message
0-9	Alpha digits
a-z	alpha character
other	special character

- e. WAP to find the age group on the basis of age.

Age	Group
0-10	Child
10-19	teenage
19-40	young
Above 40	old

## **Switch case statement:**

**Switch statement tests the value of a variable and compares it with multiple cases. Once the case match is found, a block of statements associated with that particular case is executed.**

```
switch( expression )  
{
```

**case value-1:**

**Block-1;**  
**Break;**

**case value-2:**

**Block-2;**  
**Break;**

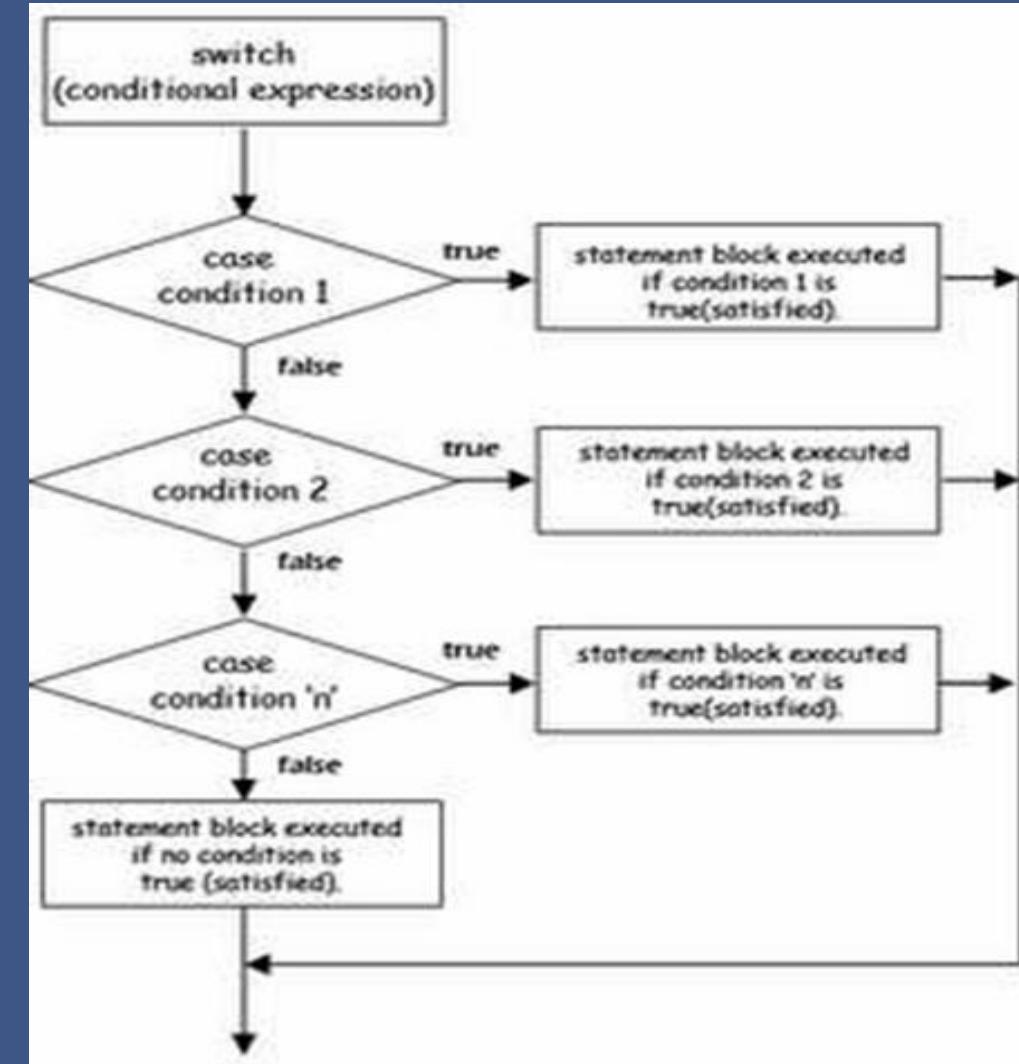
**case value-n:**

**Block-n;**  
**Break;**

**default:**

**Block-1;**  
**Break;**

```
}  
Statement-x;
```



**A program to display the name of the day in a week depending on the number, entered through the keyboard using the switch-case statement.**

```
#include<stdio.h>
#include<conio.h>
void main()
{
    clrscr(); int day;
    printf("Enter number any number from 1 to 7 \n");
    scanf("%d",&day);
    switch(day)
    {
        case 1: printf("Sunday"); break;
        case 2:
            printf("\n Monday"); break;
        case 3:
            .....
            .....
        case 7:
            printf("\n Saturday"); break;
        default:
            printf("\n Invalid entry");
    }
    getch();
}
```

## **Switch case statement related question.**

- a. Write a program that accepts a single character and give output using switch case.**

<b>Character</b>	<b>Color</b>
R or r	Red
B or b	Blue
W or w	White
Otherwise	no more priority color

- b. Write a menu driven program to perform following: adding two numbers, difference, product, remainder when one number is divided by another.**

**‘OR’**

**Write a program that accepts two integer and an operator (+,-,\* ,/,%). And the program should calculate the value as directed by the operator and display the result (use switch case)**

**Looping :**

**Looping in the processing of solving required problem with conduction and logic repeatedly.**

**Purpose of loop**

- a) Execute statements for a specified number of times.
- b) To use a sequence of values.

**Types of loop**

- a) For loop
- b) While loop
- c) Do while loop

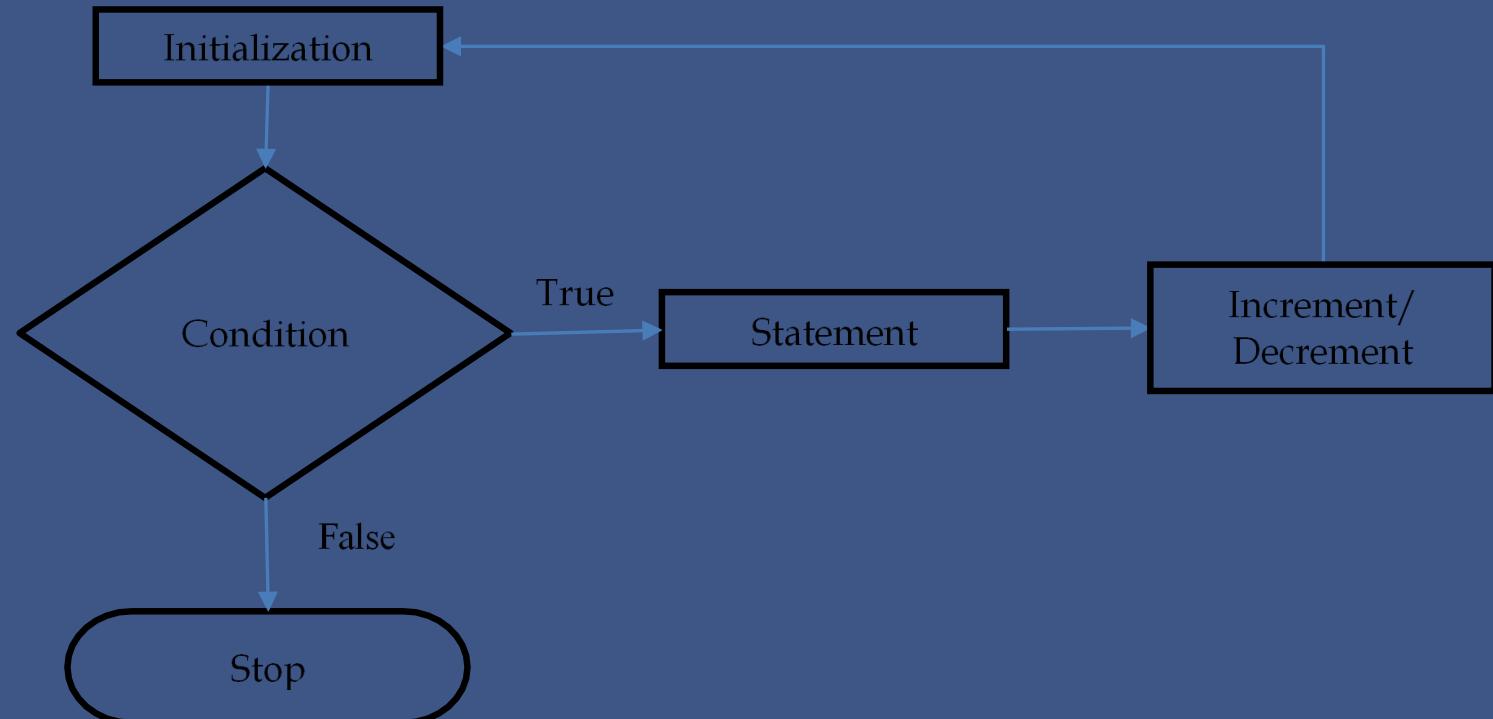
## For Loop

The for loop is the most commonly used statement in c . This loop consist of three expressions. The first expression issued to initialize the index value, the second to check whether the loop is to be continued again and third to change the index value for further repetition.

Syntax:

**For(Initialization; Condition; increment/decrement)**

{  
Statement;  
}



**WAP to display “c programming “ ten times.**

```
#include<stdio.h>
#include<conio.h>
Void main() / int main()
{
    Clrscr();
    int i;
    for(i=1;i<=10;i++)
        //for loop to print c programming 10 times
    {
        printf("\nC programming");
    }
    getch();
}
```

**Write a program to find sum of first n natural number using for loop.**

```
#include<stdio.h>
#include<conio.h>
void main()
{
int i,n,s=0;
printf("Enter the value of n ");
scanf("%d",&n);
    for(i=1;i<=n;i++)
{
s=s+i;
}
printf("The sum upto %d is %d.",s);
getch();
}
```

**WAP to fine the sum of 1 to 100 number using for loop.**

. Write a program to read two number n1 and n2. Display all even numbers between n1 and n2.

```
#include<stdio.h>
#include<conio.h>
void main()
{
int i,n1,n2;
printf("Enter the value of n1 and n2 ");
scanf("%d%d",&n1,&n2);
for(i=n1;i<=n2;i++)
{
if(i%2==0)
printf("%d\t",i);
}
getch();
```

- 1. Write a program to find the greatest number among ‘n’ integer entered from the keyboard.**
- 2. Write a program to display Multiplication Table using for loop.**
- 3. Write a program to find the sum of odd and even number for 1 to 50.**
- 4. Write a program to find factorial of a given number.**
  
- 5. Write a program to display Fibonacci series upto 10th terms.**  
0   1   1   2   3   5   8   13 .... upto 10th terms
- 6. Write a program to read ‘n’ number, find sum of positive and sum of negative numbers.**

Write a program to find sum of first n natural number using for loop.

```
#include<stdio.h>
#include<conio.h>
void main()
{
int i,n,s=0;
printf("Enter the value of n ");
scanf("%d",&n);
for(i=1;i<=n;i++)
{
s=s+i;
}
printf("The sum upto %d is %d.",s);
getch();
}
```

**Write a program to find the sum of 1 to 100 number using for loop.**

```
#include<stdio.h>
#include<conio.h>
void main()
{
int i,n=100,s=0;
for(i=1;i<=n;i++)
{
s=s+i;
}
printf("The sum upto %d is %d.",s);
getch();
}
```

**Write a program to display Multiplication Table using for loop.**

```
#include<stdio.h>
#include<conio.h>
void main()
{
    int i,n;
    printf("Enter number ");
    scanf("%d",&n);
    for(i=1;i<=10;i++)
    {
        printf("\n% d*%d \t = \t %d",n,i,n*i);
    }
    getch();
}
```

**Write a program to find factorial of a given number**

```
#include<stdio.h>
#include<conio.h>
void main()
{
int i,n,fact=1;
printf("Enter any number ");
scanf("%d",&n);
for(i=n;i>=1;i--)
{
fact=fact*i;
}
printf("\n The factorial of %d = %d ",n,fact);
getch();
}
```

Write a program to display the following series using for loop.

Write a program to display Fibonacci series upto 10th terms.

0      1      1      2      3      5      8      13 .... upto 10th terms

```
#include<stdio.h>
#include<conio.h>
void main()
{
int a=0,b=1,c,i;
printf("%d\t%d\t",a,b);
for(i=1;i<=8;i++)
{
c=a+b;
printf("%d\t",c);
a=b;
b=c;
}
getch();
}
```

Q. Write a program to read 'n' number, find sum of positive and sum of negative numbers.

```
#include<stdio.h>
#include<conio.h>
void main()
{
int i,s_pos=0,s_neg=0,n,no;
printf("How many numbers ");
scanf("%d",&no);
for(i=1;i<=no;i++)
{
printf("\n Enter number ");
scanf("%d",&n);
if(n>0)
s_pos=s_pos+n;
Else
s_neg=s_neg+n;
}
printf("\n The sum of positive numbers = %d ",s_pos);
printf("\n The sum of negative numbers = %d ",s_neg);
getch();
```