

**VI Relevant Affective domain related Outcome(s)**

1. Follow safety practices.
2. Practice optimal way to solve problem.

**VII Minimum Theoretical Background**

Concept of flowcharts and algorithms, symbols of flowchart, guidelines for preparing flowchart and algorithms, concept of variables and constants, knowledge of arithmetic operators.

**VIII Algorithm**

Input : Two numbers

Output : Addition of two numbers

Step 1 : Start

Step 2 : Declare variable num1, num2 and Add

Step 3 : Read value for num1 and num2

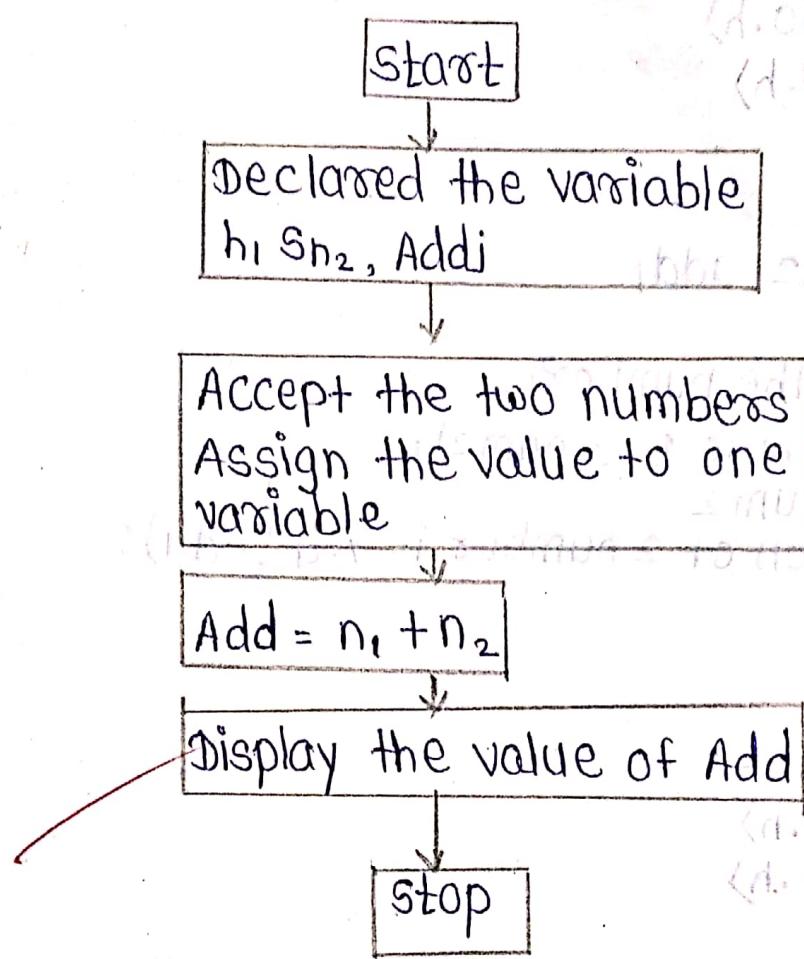
Step 4 : find num1 & num2 & assign the result  
too add

$$ADD = NUM1 + NUM2$$

Step 5 : Display value of ADD

Step 6 : Stop

## IX Flow Chart



## X C Program Code

\* #include <stdio.h>  
#include <conio.h>  
void main()  
{  
int num1, num2, add;  
clrscr();  
printf("enter the numbers");  
scanf("%d,%d", &num1, &num2);  
add = num1 + num2;  
printf("addition of 2 number is %d", add);  
getch();  
}

\* #include <stdio.h>  
#include <conio.h>  
int main()  
{  
int c, d, temp;  
clrscr();  
printf("enter the value of c & d\n");  
scanf("%d %d", &c, &d);  
printf("before interchange\n c=%d \n d=%d \n", c, d);  
temp = c;  
c = d;  
printf("After interchange\n c=%d  
d=%d \n", c, d);  
getch();  
}

**XI Resources required**

Sr.No.	Name of Resource	Major Specification	Qty.	Remarks
1.	Computer System	Any desktop or laptop computer with basic configuration	One computer system for each student	
2.	'C' Compiler	Turbo C/gcc	One for each computer system	

**XII Precautions to be followed**

- Handle computer system with care.
- Strictly follow the instruction for writing, compiling and executing the program.
- Start and Shutdown system with proper procedure.

**III Resources used**

Sr. No.	Name of Resource	Specifications		Remarks (If any)
		Make	Details	
1.	Computer System with broad specifications		Windows 10 is processor 2 GB RAM	
2.	Software		Turbo C	
3.	Any other resource used			

**IV Results****V Conclusion**(Actions/decisions to be taken based on the interpretation of results).

Thus we written 'C' program using constant variable and mathematical expression.

**VI Practical Related Questions**

Note: Below given are few sample questions for reference. Teachers must design more such questions so as to ensure the achievement of identified CO.

- Write error message given by C Compiler during program compilation, if variable is used without declaration.
- Write error message given by C Compiler during program compilation, if you use %d to print float variable.
- Write error message given by C Compiler during program compilation, if you don't use '&' sign in scanf statement.
- Write output, if '&' sign with variable name is given while printing the output.

## Space for Answer

```
* #include <stdio.h>
#include <conio.h>
void main()
{
    float r, val, area;
    printf("enter radius value");
    scanf("%d", &r);
    Area = 4 * 3.14 * r * r;
    vol = 4 * 3 * 3.14 * r * r * r;
    printf("Area of sphere is %f", area);
    printf("Volume of sphere is %f", vol);
    getch();
}
```

*Q. What is addition?*

```
* #include <stdio.h>
#include <conio.h>
void main()
{
    int num1, num2, num3, num4, num5, add;
    clrscr();
    printf("enter the number");
    scanf("%d,%d,%d,%d,%d", &num1, &num2, &num3,
        &num4, &num5);
    add = num1 + num2 + num3 + num4 + num5;
    printf("addition of 5 number is %d", add);
    getch();
}
```

## \* Output

① undefined symbol

② NO errors

③ NO error

④ It shows the address instead of value.

## Practical No. 2: C programs using increment/decrement operators and data type conversion.

*Note: Teachers are advised to prepare programming problems on similar guidelines as given below. Each student should develop and execute minimum two programs.*

1. Write a program to take one integer number as input, assign its value to another variable with pre and post increment operator and display its value.
2. Write a program for pre and post decrement operator.
3. Write a program for Implicit and Explicit data type conversion.

### I Practical Significance

This practical is useful for student to understand the use of increment/decrement operators. After the completion of this practical student will be able to use increment/decrement operators and convert data from one type to another.

### II Relevant Program Outcomes (POs)

- **Basic knowledge:** Apply knowledge of basic mathematics, sciences and basic engineering to solve the broad-based Computer engineering problem.
- **Discipline knowledge:** Apply Information Technology knowledge to solve broad-based Information Technology related problems.
- **Experiments and practice:** Plan to perform experiments, practices and to use the results to solve Information Technology related problems.
- **Engineering tools:** Apply appropriate Information Technology related techniques/tools with an understanding of the limitations.

### III Competency and Practical skills

This practical is expected to develop the following skills for the industry identified competency '*Develop 'C' programs to solve broad-based computer related problem*' :

1. Write Algorithm for given program.
2. Draw Flow Chart for given Algorithm.
3. Develop a 'C' program using variables, constants and arithmetic expressions.
4. Debug the program.
5. Execute a simple C program using Turbo C/ gcc compiler.

### IV Relevant Course Outcome(s)

1. Develop flowchart and algorithm to solve problems logically.
2. Write simple 'C' programs using arithmetic expressions.

### V Practical Outcome

1. Write/compile/execute simple 'C' program using Turbo 'C' /gcc compiler.
2. Write simple 'C' program to exhibit increment/decrement operators and data type conversion.

### VI Relevant Affective domain related Outcome(s)

1. Follow safety practices.

**VII Minimum Theoretical Background**

Concept of increment and decrement operators, hierarchy of operators, different data types and type conversion.

**VIII Algorithm**

- Step 1 : Start
- Step 2 : declare the variable  $n_1, dn_2$
- Step 3 : Print the statement enter the number
- Step 4 : Give the condition  $n_2 = ++n_1$  of pre. increment
- Step 5 : Give the condition  $n_2 = n_1++$  of post. increment
- Step 6 : enter the  $n_1$  to get the value of  $n_2$ .
- Step 7 : print the output
- Step 8 : stop

2.

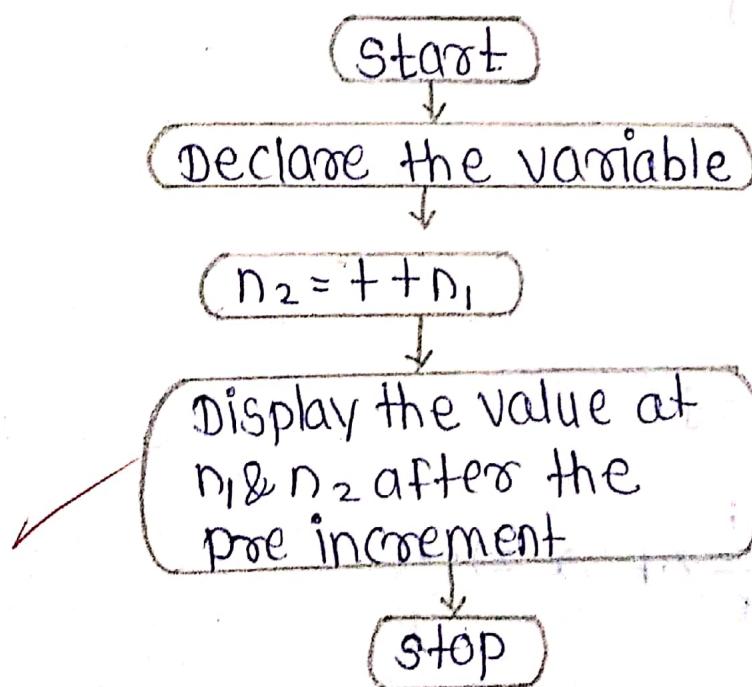
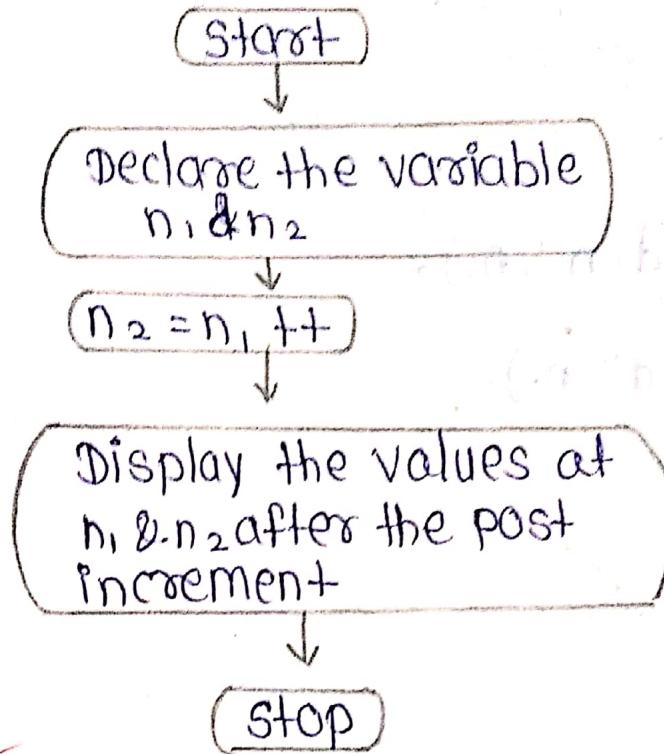
- Step 1 : start
- Step 2 : declare the variable  $n_1 \& n_2$
- Step 3 : Print the statement enter the numbers
- Step 4 : Give conditions  $n_2 = --n_1$  &  $n_2 = n_1--$
- Step 5 : Enter  $n_1$  to get the value of  $n_2$
- Step 6 : print output
- Step 7 : stop

**for implicit**

- Step 1 : Start
- Step 2 : declare int & char type of data x & y.
- Step 3 : if x is int it will convert to float and y will int if y = char
- Step 4 : print the value of float z & x.

**for explicit**

- Step 1 : start
- Step 2 : declare these for implicit if double.
- Step 3 : equation int sum = (int)
- Step 4 : print the value of x
- Step 5 : stop



## X 'C' Program Code

1. #

```
int main()
{
    int n, = 1;
    n2 = n2 + +
    printf("1.d, n", n2);
    n2 = ++n, ;
    printf("2.d", n2);
    getch();
}
```

2. #

```
#include <stdio.h>
int main()
{
    int n, = 2;
    n2 = -n1;
    printf("1.d", n2);
    getch();
}
```

3. implicit

```
#include <stdio.h>
int main()
{
    int x = 10;
    char y = 'a'; // a = 97
    x = x + y;
    float z = x + 10;
    printf("x = %d, z = %f", x, z);
    return();
}
```

4. explicit

```
#include <stdio.h>
int main()
{
    double x =
    int sum = (int
    printf("sum =
    return();
}
```

## XI Resources required

S.No.	Name of Resource	Major Specification	Qty.	Remarks
1.	Computer System	Any desktop or laptop computer with basic configuration	One computer system for each student	
2.	'C' Compiler	Turbo C/gcc	One for each computer system	

## XII Precautions to be followed

1. Handle computer system with care.
2. Strictly follow the instruction for writing, compiling and executing the program.
3. Start and Shutdown system with proper procedure.

## XIII Resources used

S.No.	Name of Resource	Specifications		Remarks (If any)
		Make	Details	
1.	Computer System with broad specifications		window 10 i5 processor 26m	
2.	Software		Turbo C	
3.	Any other resource used			

## XIV Results

.....  
.....

## XV Conclusion (Actions/decisions to be taken based on the interpretation of results).

Thus we written C program using increment, decrement operators & data type conversion.....

## XVI Practical Related Questions

Note: Below given are few sample questions for reference. Teachers must design more such questions so as to ensure the achievement of identified CO.

1. Write the output of following C Program.

```
#include<stdio.h>
int main()
{
    int i=1;
    i=2+2*i++;
    printf("%d",i);
    return 0;
```

}

2. Write the output of following C Program

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

```
int main()
{
    int i = 2;
    int j = ++i + i;
    printf("%d\n", j);
}
```

3. If  $x=7$  and  $y=-3$ , write values of  $z$  for following C Statement.

```
z = ++x + y - - - ++y - x - - x - - ++y - x - -;
```

### Space for Answer

i) Output :- 5

ii) Output :- 4

### VII References / S

1. <https://www>
2. <http://spok>  
search/?s

### III Assessment

1.	Deb
2.	Foll
3.	Con
4.	Qua
5.	Con
6.	Ans

### Process Related(7.5)

5

## Practical No. 3: Write simple C program to convert temperature in Fahrenheit degrees to Centigrade degrees.

### I Practical Significance

This practical is useful for students to understand the conversion of temperature from Fahrenheit degrees to Centigrade degrees. After the completion of this practical student will be able to use arithmetic formulas for writing programs.

### II Relevant Program Outcomes (POs)

- **Basic knowledge:** Apply knowledge of basic mathematics, sciences and basic engineering to solve the broad-based Computer engineering problem.
- **Discipline knowledge:** Apply Information Technology knowledge to solve broad based Information Technology related problems.
- **Experiments and practice:** Plan to perform experiments, practices and to use the results to solve Information Technology related problems.
- **Engineering tools:** Apply appropriate Information Technology related techniques tools with an understanding of the limitations.

### III Competency and Practical skills

This practical is expected to develop the following skills for the industry identified competency '*Develop 'C' programs to solve broad-based computer related problem*'.

1. Write Algorithm for given program.
2. Draw Flow Chart for given Algorithm.
3. Develop a 'C' program using variables, constants and arithmetic expressions.
4. Debug the program.
5. Execute a simple C program using Turbo C/ gcc compiler.

### IV Relevant Course Outcome(s)

1. Develop flowchart and algorithm to solve problems logically.
2. Write simple 'C' programs using arithmetic expressions.

### V Practical Outcome

1. Write/compile/execute simple 'C' program using Turbo 'C' / gcc compiler.
2. An ability to use and convert mathematical expressions into C statements.

### VI Relevant Affective domain related Outcome(s)

1. Maintain tools and equipment.
2. Follow ethical practices.

### VII Minimum Theoretical Background

Concept of various arithmetic expressions and use of formulas for conversion and hierarchy of operators.

**VIII Algorithm**

Step 1: start

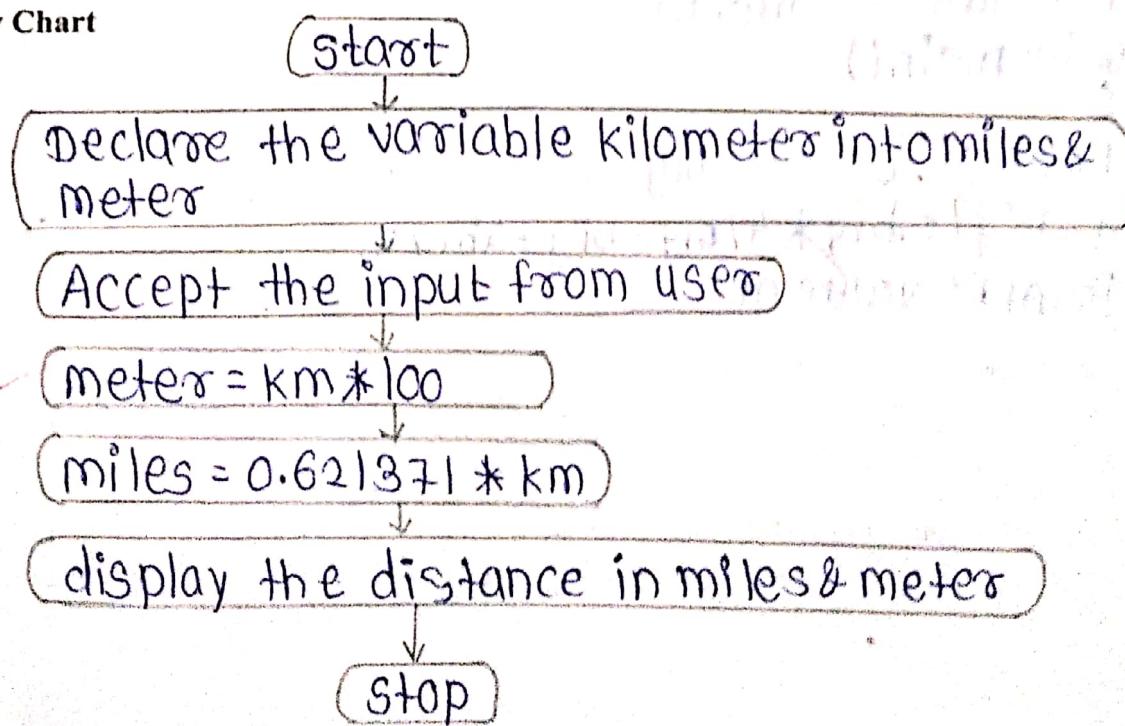
Step 2: Declare the var km, meter & miles

Step 3: Accept the i/p value from meter

Step 4: perform conversion separation meter=km\*100;

Step 5: display the distance in meters & miles

Step 6: Stop

**IX Flow Chart**

## X 'C' Program Code

1] #include <stdio.h>  
 #include <conio.h>  
 void main()  
 {  
 int a, b, c, y, z, p;  
 clrscr();  
 printf("enter the value of a, b, c");  
 scanf("%d,%d,%d", &a, &b, &c);  
 y = b \* b;  
 z = 24 \* a \* c;  
~~p = z \* a;~~  
~~11x = -b \* y \* z / 11;~~  
 printf("value of its %.d")  
 getch();  
 }

2] #include <stdio.h>  
 #include <conio.h>  
 void main()  
 {  
 float g, abc = 2.5, big = 2;  
~~g = big / 2 + big \* 4 / big - big + abc / 3;~~  
 printf("value of")

## XI Resources required

S.No.	Name of Resource	Major Specification	Qty.	Remarks
1.	Computer System	Any desktop or laptop computer with basic configuration	One computer system for each student	
2.	'C' Compiler	Turbo C/gcc	One for each computer system	

## **XII      Precautions to be followed**

- Handle computer system with care.
  - Strictly follow the instruction for writing, compiling and executing the program.
  - Start and Shutdown system with proper procedure.

### XIII Resources used

S.No.	Name of Resource	Specifications		Remarks (If any)
		Make	Details	
1.	Computer System with broad specifications		Windows 10 i5 processor 2GB RAM	
2.	Software		TURBO C	
3.	Any other resource used			

## XIV Results

## XV Conclusion (Actions/decisions to be taken based on the interpretation of results)

Then we written simple c program to convert temperature in farbenit degree to centigrade degree.

## **XVI Practical Related Questions**

*Note: Below given are few sample questions for reference. Teachers must design more such questions so as to ensure the achievement of identified CO.*

1. Convert the following mathematical formula into appropriate C statements.  
$$X = \frac{-b(b * b) - 24ac}{2a}$$
  2. Point out the error in following statement.  
$$k = (a * b)(c + (2.5a + b)(d + e));$$
  3. Evaluate the following expressions and show its hierarchy.  
$$g = big / 2 + big * 4 / big - big + abc / 3;$$

(abc = 2.5, big = 2, assume g to be a float)

4. Write a C program to convert Kilometer into Miles and Meters.
5. Write a C program to convert temperature in Centigrade degrees to Kelvin degrees.

### Space for Answer

```
#include <stdio.h>
#include <conio.h>
void main()
{
    float km, meter, miles;
    clrscr();
    printf("1. F.", &km);
    meter = km * 1000;
    miles = 0.621371 * km;
    printf("The equivalent distance in meters is:");
    scanf("1. F. 1. F.", meter, miles);
    getch();
}
```

## **Practical No. 4: Write simple programs to calculate the area and perimeter of the rectangle, and the area & circumference of the circle.**

### **I Practical Significance**

Student will be able to use constants, different data types, modulus operator and arithmetic function to find area and circumference of different geometric figures.

### **II Relevant Program Outcomes**

- **Basic knowledge:** Apply knowledge of basic mathematics, sciences and basic engineering to solve the broad-based Computer engineering problem.
- **Discipline knowledge:** Apply Information Technology knowledge to solve broad-based Information Technology related problems.
- **Experiments and practice:** Plan to perform experiments, practices and to use the results to solve Information Technology related problems.
- **Engineering tools:** Apply appropriate Information Technology related techniques tools with an understanding of the limitations.

### **III Competency and Practical skills**

This practical is expected to develop the following skills for the industry identified competency '*Develop 'C' programs to solve broad-based computer related problem*' :

1. Write Algorithm for given program.
2. Draw Flow Chart for given Algorithm.
3. Develop a 'C' program using variables, constants and arithmetic expressions.
4. Debug the program.
5. Execute a simple C program using Turbo C/ gcc compiler.

### **IV Relevant Course Outcome(s)**

1. Develop flowchart and algorithm to solve problems logically.
2. Write simple 'C' programs using arithmetic expressions.

### **V Practical Outcome**

1. Write/compile/execute simple 'C' program using Turbo 'C' / gcc compiler.
2. To generate an ability to use constants and default values in the program.

### **VI Relevant Affective domain related Outcome(s)**

1. Practice optimal way to solve problem.
2. Demonstrate working as a leader/a team member.

### **VII Minimum Theoretical Background**

Concept of constants and their default values, concept of modulus operator and mathematical functions.

## VIII Algorithm

## Rectangle

Algorithm : Area & Perimeter of Rectangle

Input : length & breadth of the rectangle

Output : Area & Perimeter of rectangle

Step 1 : Start

Step 2 : Declare Variable LENGTH, WIDTH & AREA & PERIMETERS

Step 3 : Read values for Length & Width

Step 4 : Calculate area & perimeter of rectangle & assign the result to area & perimeter of rectangle & assign the result to area & perimeter of rectangle

Step 5 : Display value of AREA & PERIMETER

Step 6 : Stop

## Circle

Algorithm : Area & Circumference of circle

Input : radius of the circle

Output : Area & Circumference of circle

Step 1 : Start

Step 2 : Declare Variable RADIUS, AREA & CIRCUMFERENCE

Step 3 : Read values for R

Step 4 : Calculate area & circumference of circle and assign the result to the area of circumference

Step 5 : Display value of Area & circumference

Step 6 : Stop

## IX Flow Chart

## Rectangle

Start

Declare the variables l, b, Perimeter, area

Accept the values l & b

$$\text{Area} = l \times b$$

$$\text{Perimeter} = 2(l+b)$$

Display the value of area & perimeter

Stop

## circle

Start

Declare the variables r, area & circumference

Accept the value of r

$$\text{Area} = \pi r^2$$

$$\text{Circumference} = 2\pi r$$

Display the value of area of circumference of circle

Stop

## X 'C' Program Code

Area & perimeter of Rectangle

```
#include <stdio.h>
#include <conio.h>
void main()
{
    int l, b, area, perimeter;
    clrscr();
    printf("enter the value of length, breadth");
    scanf("%d,%d", &l, &b);
    area = l * b;
    perimeter = 2 * (l + b);
    printf("the area of rectangle is %d", area);
    printf(" the perimeter of rectangle is %d" perimeter);
    getch();
}
```

## Area & circumference of circle

```
#include <stdio.h>
#include <conio.h>
void main()
{
    int area, circumference;
    clrscr();
    printf("enter the value of radius");
    scanf("%d", &r);
    area = 3.14 * r * r;
    circumference = 2 * 3.14 * r;
    printf("the area of circle is %d", area);
    printf(" the circumference of the circle %d", circumference);
    getch();
}
```

**XI Resources required**

S.No.	Name of Resource	Major Specification	Qty.	Remarks
1.	Computer System	Any desktop or laptop computer with basic configuration	One computer system for each student	
2.	'C' Compiler	Turbo C/gcc	One for each computer system	

**XII Precautions to be followed**

- Handle computer system with care.
- Strictly follow the instruction for writing, compiling and executing the program.
- Start and Shutdown system with proper procedure.

**III Resources used**

S.No.	Name of Resource	Specifications		Remarks (If any)
		Make	Details	
1.	Computer System with broad specifications		windows 10 i5 processor 2 GB RAM	
2.	Software		Turboc	
3.	Any other resource used			

**IV Results**

.....  
.....

**XV Conclusion (Actions/decisions to be taken based on the interpretation of results).**

.....thus.....written.....simple.....C.....program.....to.....calculate.....  
.....area.....&.....perimeter.....of.....rectangle.....&.....area.....  
.....circumference.....of.....circle.....

**VI Practical Related Questions**

*Note: Below given are few sample questions for reference. Teachers must design more such questions so as to ensure the achievement of identified CO.*

- Write mathematical function to find square of a number.
- Write comments in above C program.
- Write a program to interchange the values of two variables.
- Write a program to reverse the given number.

## Space for Answer

1) → Pow() function in c is used to find the power of the given number.

"math.h" header file supports pow() functions (b language syntax for pow).

#include <stdio.h>

#include <math.h>

void main()

{

int main()

{

printf("2 power 4 = %.f \n", pow(2, 0.4, 0));

printf("5 power 3 = %.f \n", pow(5, 3));

return 0;

getch();

}

ANSWER

.....

.....

.....

.....

.....

.....

## Practical No. 5: Write C programs for decision making and branching using if and if-else structure.

1. Write a C program to determine whether a given year is a leap year or not.
2. Write a C program to determine whether a string is palindrome.

### I Practical Significance

The ability to control the normal flow of a program to make decisions on what code to be executed is valuable to the programmer. One of the important functions of the *if* statement is that it allows you to control if a program enters a section of code or not based on whether a given condition is true or false. After the completion of this practical student will be able to use decision making conditional and branching statements to solve the given problem.

### II Relevant Program Outcomes (POs)

- Basic knowledge:** Apply knowledge of basic mathematics, sciences and basic engineering to solve the broad-based Computer engineering problem.
- Discipline knowledge:** Apply Information Technology knowledge to solve broad-based Information Technology related problems.
- Experiments and practice:** Plan to perform experiments, practices and to use the results to solve Information Technology related problems.
- Engineering tools:** Apply appropriate Information Technology related techniques tools with an understanding of the limitations.
- Ethics:** Apply ethical principles for commitment to professional ethics responsibilities and norms of the practice also in the field of Computer engineering.
- Individual and team work:** Function effectively as a leader and team member in diverse/ multidisciplinary teams.
- Communication:** Communicate effectively in oral and written form.
- Life-long learning:** Engage in independent and life-long learning activities in the context of technological changes in the Computer engineering field and allied industry.

### III Competency and Practical skills

This practical is expected to develop the following skills for the industry identified competency '**Develop 'C' programs to solve broad-based computer related problem**':

1. Write Algorithm for given program.
2. Draw Flow Chart for given Algorithm.
3. Develop a 'C' program using variables, constants and arithmetic expressions.
4. Debug the program.
5. Execute a simple C program using Turbo C/ gcc compiler.

### IV Relevant Course Outcome(s)

1. Develop flowchart and algorithm to solve problems logically.
2. Develop 'C' programs using control structure.

**V Practical Outcome**

1. Write/compile/execute simple 'C' program using Turbo 'C' /gcc compiler.
2. Decision Making and branching using if and if-else structure.

**VI Relevant Affective domain related Outcome(s)**

1. Maintain tools and equipment.
2. Follow ethical practices.

**VII Minimum Theoretical Background**

Concept variable, constant, keywords and data types. Concept of relational and logical operators, if statement, if else statement, nested if-else, if-else ladder.

**III Algorithm**

Step 1 : Start

Step 2 : Declare the variable years as int

Step 3 : Accept input from user

Step 4 : Check,  $\text{year} \% = 0$

    if above statement is true then

        check  $\text{year} \% 100 = 0$  if it is

            false Year is not leap year

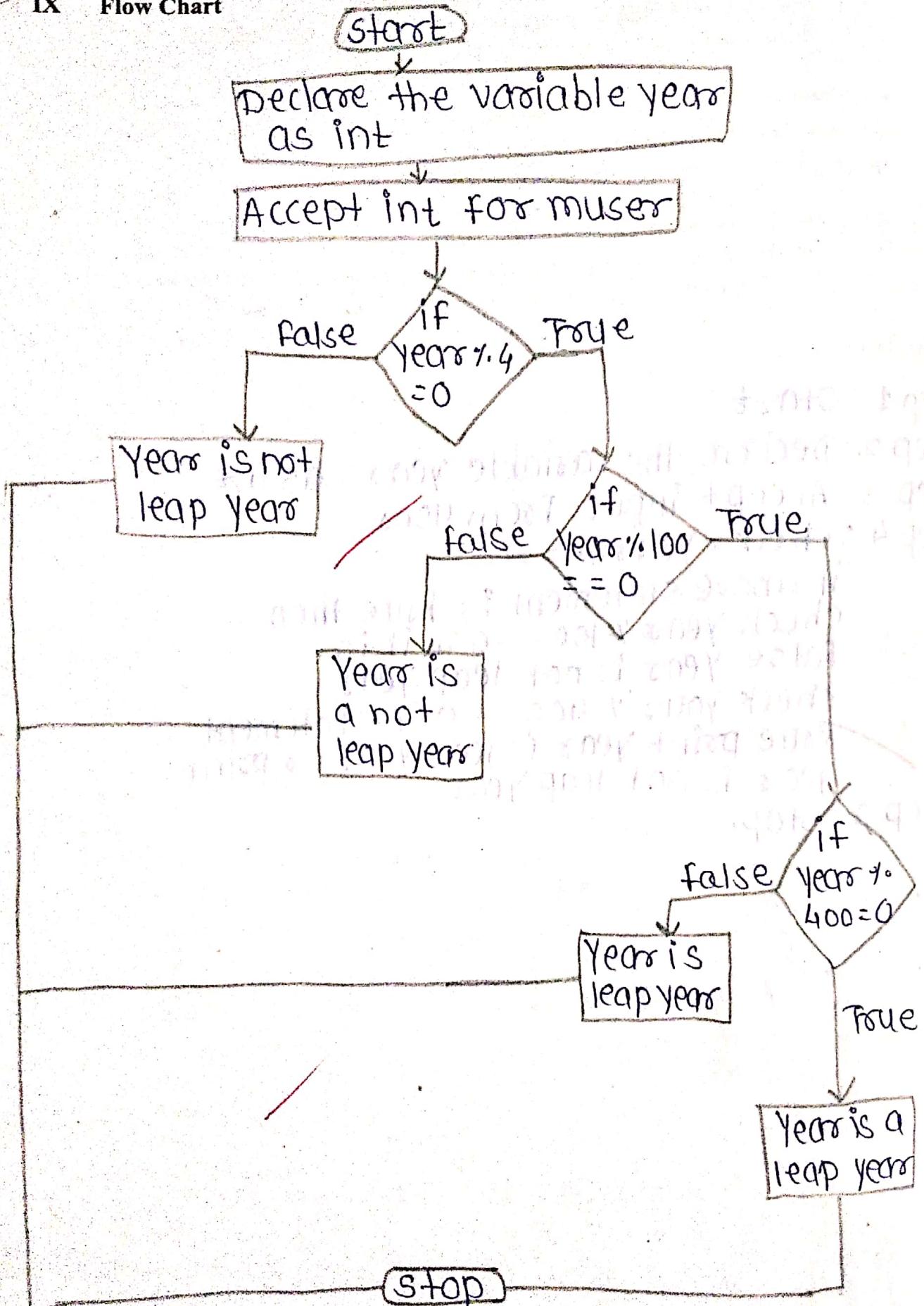
            check  $\text{year} \% 400 = 0$  if statement

                true print year is leap year or print

                year is not leap year

Step 5 : Stop.

## IX Flow Chart



## X 'C' Program Code

```
#include <stdio.h>
int main()
{
    int year;
    printf("enter a year to check if it is a leap year");
    scanf("%d", &year);
    if (year % 400 == 0)
    {
        if (year % 100 == 0)
        {
            // year is divisible by 400, hence the year is a leap year
            if (year % 400 == 0)
                printf("%d is a leap year", year);
            else
                printf("%d is not a leap year", year);
        }
        else
            printf("%d is a leap year", year);
    }
    else
        printf("%d is not a leap year", year);
    return 0;
}
```

~~problem with logic for century leap year. It is not printing century leap years.~~

**XI Resources required**

S.No.	Name of Resource	Major Specification	Qty.	Remarks
1.	Computer System	Any desktop or laptop computer with basic configuration	One computer system for each student	
2.	'C' Compiler	Turbo C/gcc	One for each computer system	

**XII Precautions to be followed**

1. Handle computer system with care.
2. Strictly follow the instruction for writing, compiling and executing the program.
3. Start and Shutdown system with proper procedure.

**XIII Resources used**

S.No.	Name of Resource	Specifications		Remarks (If any)
		Make	Details	
1.	Computer System with broad specifications		Windows 10 i3 processor 2GB RAM	
2.	Software		Turbo C	
3.	Any other resource used			

**XIV Results****XV Conclusion** (Actions/decisions to be taken based on the interpretation of results).

Thus we written C program for decision making & branching using if & if...else structure.....

**XVI Practical Related Questions**

*Note: Below given are few sample questions for reference. Teachers must design more such questions so as to ensure the achievement of identified CO.*

1. Write output of following program.

```
main()
{
    int a = 500, b, c ;
    if ( a >= 400 )
        b = 300 ;
```

```

    c = 200 ;
    printf( "\n%d %d", b, c );
}

```

2. Point out the errors, if any, in the following program.

```

main()
{
    int x = 30 , y = 40 ;
    if ( x == y )
        printf( "x is equal to y" );
    elseif ( x > y )
        printf( "x is greater than y" );
    elseif ( x < y )
        printf( "x is less than y" );
}

```

3. Write output of following program.

```

main()
{
    int k = 35 ;
    printf( "\n%d %d %d", k == 35, k = 50, k > 40 );
}

```

**Space for Answer** 2. Write a C program to determine whether a string is palindrome.

```

#include <stdio.h>
#include <string.h>
void main()
{
    char a[100], b[100];
    clrscr();
    printf("enter the string");
    gets(a);
    strcpy(b,a);
    strrev(b);
    if(strcmp(a,b) == 0)
    {
        printf("string is palindrome");
    }
    else
    {
        printf("string is not palindrome");
    }
}

```

```

3.....  

getch();  

}

```

**XVII References / Suggestions for further Reading**

1. <https://www.w3schools.in/c-tutorial/operators/>
2. <https://www.programiz.com/c-programming/c-decision-making-loops-examples>
3. [http://spoken-tutorial.org/tutorial-search/?search\\_foss=C+and+Cpp&search\\_language=English](http://spoken-tutorial.org/tutorial-search/?search_foss=C+and+Cpp&search_language=English)

**XVIII Assessment Scheme**

Performance indicators		Weightage
<b>Process related (7.5 Marks)</b>		<b>30%</b>
1.	Debugging ability	20%
2.	Following ethical practices	10%
<b>Product related (17.5 Marks)</b>		<b>70%</b>
1.	Correctness of algorithm	15%
2.	Correctness of Flow chart	15%
3.	Correctness of Program codes	20%
4.	Quality of input messaging and output formatting	5%
5.	Completion and submission of Practical in time	5%
6.	Answer to sample questions	10%
<b>Total (25 Marks)</b>		<b>100%</b>

Marks obtained			Dated Sign of Teacher
Process Related(7.5)	Product Related(17.5)	Total(25)	
5	15	20	Shubhangi

### Practical No. 6: Write program to:

- 1) Find the greatest of the three numbers using conditional operators.
- 2) Find if a given character is vowel.

#### I Practical Significance

Students will be able to write programs using conditional operators to solve the given problem.

#### II Relevant Program Outcomes

- **Basic knowledge:** Apply knowledge of basic mathematics, sciences and basic engineering to solve the broad-based Computer engineering problem.
- **Discipline knowledge:** Apply Information Technology knowledge to solve broad-based Information Technology related problems.
- **Experiments and practice:** Plan to perform experiments, practices and to use the results to solve Information Technology related problems.
- **Engineering tools:** Apply appropriate Information Technology related techniques/tools with an understanding of the limitations.
- **Ethics:** Apply ethical principles for commitment to professional ethics, responsibilities and norms of the practice also in the field of Computer engineering.
- **Individual and team work:** Function effectively as a leader and team member in diverse/ multidisciplinary teams.
- **Communication:** Communicate effectively in oral and written form.
- **Life-long learning:** Engage in independent and life-long learning activities in the context of technological changes in the Computer engineering field and allied industry.

#### III Competency and Practical skills

This practical is expected to develop the following skills for the industry identified competency '*Develop 'C' programs to solve broad-based computer related problem*':

1. Write Algorithm for given program.
2. Draw Flow Chart for given Algorithm.
3. Develop a 'C' program using variables, constants and arithmetic expressions.
4. Debug the program.
5. Execute a simple C program using Turbo C/ gcc compiler.

#### IV Relevant Course Outcome(s)

1. Develop flowchart and algorithm to solve problems logically.
2. Develop 'C' programs using control structure.

#### V Practical Outcome

1. Write/compile/execute simple 'C' program using Turbo 'C' /gcc compiler.
2. Decision Making and branching using if and if-else structure.

**VI Relevant Affective domain related Outcome(s)**

1. Follow safety practices.
2. Maintain tools and equipment.

**VII Minimum Theoretical Background**

Basic program logic using different operators and expressions.

**VIII Algorithm**

Algorithm = Greatest number using conditional operators

Input = A number

Output = Display greatest number

Step 1 = start

Step 2 = Declare variable A, B, C, big

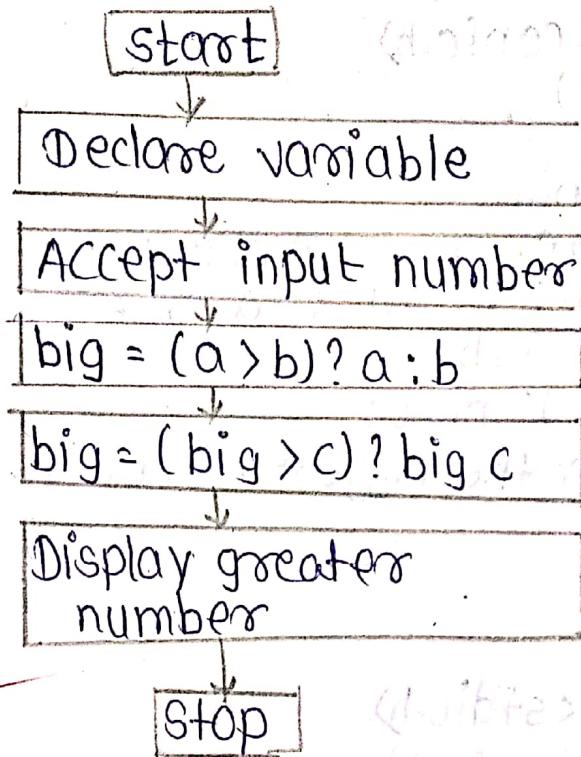
Step 3 = Read value for A, B & C

Step 4 = big = (a>b)? a : b;

                big = (big > c)? big : c;

Step 5 = Stop

## IX Flow Chart



**X 'C' Program Code**

```
#include <stdio.h>
#include <conio.h>
void main()
{
    int a, b, c, big;
    printf("enter three numbers");
    scanf("%d, %d, %d", &a, &b, &c);
    big(a>b)? a : b;
    big(b>c)? big : c;
    printf("In the biggest number is %d", big);
    getch();
}
```

```
#include <stdio.h>
#include <conio.h>
void main()
{
    char ch;
    printf("enter the character");
    scanf("%c", &ch);
    if(ch == 'a' || ch == 'A' || ch == 'e' || ch == 'E')
        printf("given character is vowel");
    else
        printf("given character is not a vowel");
    getch();
}
```

**I Resources**

S.No.
1.
2.

**I Precautions**

1. Hand
2. Strict
3. Start
4. Synta

**I Resources**

S.No.	N
1.	C
2.	S
3.	A

**Results****Conclusion**

Thus we can find the number given character is vowel or not.

**Practical Relation**

Note: Below are some more such questions.

1. Write output of the following program:
- ```
#include <stdio.h>
int main()
{
    int a, b, c;
    a = 10;
    b = 20;
    c = a + b;
    printf("a = %d, b = %d, c = %d", a, b, c);
}
```

## XI Resources required

| S.No. | Name of Resource | Major Specification                                     | Qty.                                 | Remarks |
|-------|------------------|---------------------------------------------------------|--------------------------------------|---------|
| 1.    | Computer System  | Any desktop or laptop computer with basic configuration | One computer system for each student |         |
| 2.    | 'C' Compiler     | Turbo C/gcc                                             | One for each computer system         |         |

## KII Precautions to be followed

1. Handle computer system with care.
  2. Strictly follow the instruction for writing, compiling and executing the program.
  3. Start and Shutdown system with proper procedure.
  4. Syntax for conditional operator should be given properly.

### **III Resources used**

| S.No. | Name of Resource                          | Specifications |                                        | Remarks<br>(If any) |
|-------|-------------------------------------------|----------------|----------------------------------------|---------------------|
|       |                                           | Make           | Details                                |                     |
| 1.    | Computer System with broad specifications |                | Windows 10<br>i3 processor<br>2 GB RAM |                     |
| 2.    | Software                                  |                | TURBO C                                |                     |
| 3.    | Any other resource used                   |                |                                        |                     |

## IV Results

**V Conclusion** (Actions/decisions to be taken based on the interpretation of results).

**IV Conclusion** (Actions/decisions to be taken based on the interpretation)  
Thus we written C program to find greatest of 3 numbers using conditional operators find if a given character is vowel.

## **VI Practical Related Questions**

**Practical Related Questions**  
Note: Below given are few sample questions for reference. Teachers must design more such questions so as to ensure the achievement of identified CO.

1. Write output of the following program

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

```
#include  
int main()
```

1

```
int x = 2, y = 0;
```

```
int z = (y++) ? y == 1 && x : 0;
printf("%d\n", z);
return 0;
}
```

2. Write output of following program

```
#include <stdio.h>
void main()
{
    int k = 8;
    int m = 7;
    int z = k < m ? k++ : m++;
    printf("%d", z);
}
```

3. Point out the error in the following program.

```
main( )
{
    int tag = 0, code = 1 ;
    if ( tag == 0 )
        ( code > 1 ? printf( "\nHello" ) ? printf( "\nHi" ) );
    else
        printf( "\nHello Hi !!");
}
```

Space for Answer

① 0

② 7

- ③ (i) - Declaration Syntax error  
(ii) - Declaration terminated incorrectly  
(iii) - Declaration terminated incorrectly  
(iv) Can terminate string characters

## Practical No. 7: Using switch statement

**Write programs to:**

1. Print day of week by taking number from 1 to 7.
2. Print a student's grade by accepting percent marks.

### I Practical Significance

Student will be able to understand use of switch case over if-else statement and will solve given problem using switch statement.

### II Relevant Program Outcomes

- **Basic knowledge:** Apply knowledge of basic mathematics, sciences and basic engineering to solve the broad-based Computer engineering problem.
- **Discipline knowledge:** Apply Information Technology knowledge to solve broad-based Information Technology related problems.
- **Experiments and practice:** Plan to perform experiments, practices and to use the results to solve Information Technology related problems.
- **Engineering tools:** Apply appropriate Information Technology related techniques/tools with an understanding of the limitations.
- **Ethics:** Apply ethical principles for commitment to professional ethics, responsibilities and norms of the practice also in the field of Computer engineering.
- **Individual and team work:** Function effectively as a leader and team member in diverse/ multidisciplinary teams.
- **Communication:** Communicate effectively in oral and written form.
- **Life-long learning:** Engage in independent and life-long learning activities in the context of technological changes in the Computer engineering field and allied industry.

### III Competency and Practical skills

This practical is expected to develop the following skills for the industry identified competency '*Develop 'C' programs to solve broad-based computer related problem*':

1. Write Algorithm for given program.
2. Draw Flow Chart for given Algorithm.
3. Develop a 'C' program using variables, constants and arithmetic expressions.
4. Debug the program.
5. Execute a simple C program using Turbo C/ gcc compiler.

### IV Relevant Course Outcome(s)

1. Develop flowchart and algorithm to solve problems logically.
2. Develop 'C' programs using control structure.

### V Practical Outcome

1. Write/compile/execute simple 'C' program using Turbo 'C' and gcc compiler.
2. Use switch statements in the program.

**VI Relevant Affective domain related Outcome(s)**

1. Follow safety practices.
2. Follow ethical practices.

Flow Ch

**VII Minimum Theoretical Background**

Basic program logic using different decision making and branching statements.

**VIII Algorithm**

Step 1:- Start

Step 2: Declare the variables of number

Step 3: Accept the number from user between 1 to 7

Step 4: Compare the input with switch case value

Case 1: Display Monday;

Case 2: Display Tuesday;

case 3: Display wednesday,

case 4: Display Thursday;

case 5: Display Friday;

case 6: Display Saturday;

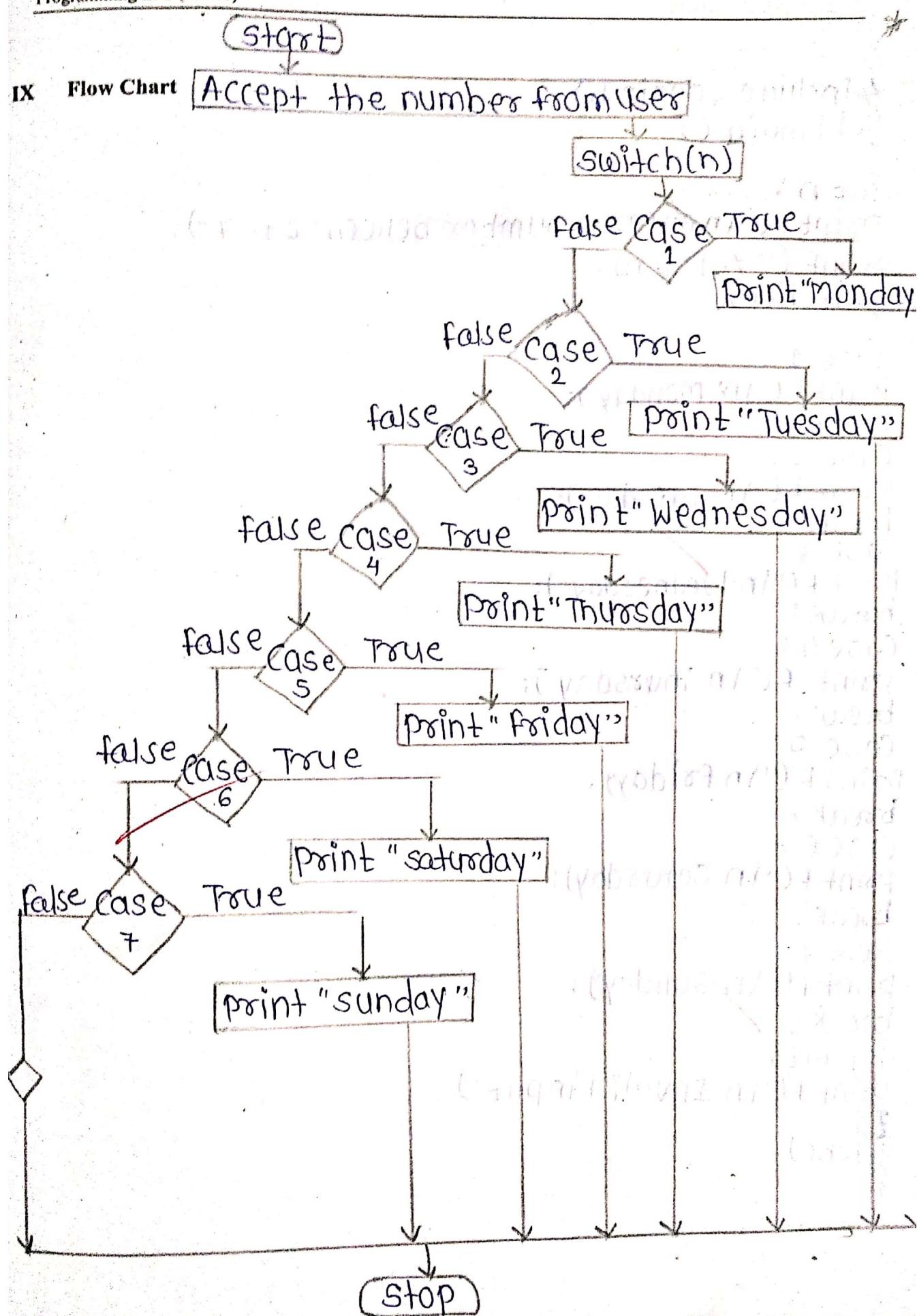
case 7: Display Sunday;

Step 5: default output invalid input

Step 6 : stop



## IX Flow Chart



## X 'C' Program Code

```

#include <conio.h>
void main ()
{
    int n;
    printf ("In enter a number between " 0 to 7 );
    scanf ("%d" & n);
    switch(n)
    {
        case 1:
            printf ("In Monday");
            break;
        case 2:
            printf ("In Tuesday");
            break;
        case 3:
            printf ("In Wednesday");
            break;
        case 4:
            printf ("In Thursday");
            break;
        case 5:
            printf ("In Friday");
            break;
        case 6:
            printf ("In Saturday");
            break;
        case 7:
            printf ("In Sunday");
            break;
        default:
            printf ("In Invalid input");
    }
    getch();
}

```

**XI Resources required**

| S.No. | Name of Resource | Major Specification                                     | Qty.                                 | Remarks |
|-------|------------------|---------------------------------------------------------|--------------------------------------|---------|
| 1.    | Computer System  | Any desktop or laptop computer with basic configuration | One computer system for each student |         |
| 2.    | 'C' Compiler     | Turbo C/gcc                                             | One for each computer system         |         |

**XII Precautions to be followed**

- Handle computer system with care.
- Strictly follow the instruction for writing, compiling and executing the program.
- Start and Shutdown system with proper procedure.
- Syntax for switch cases should be given properly.
- The case conditions must be constant, or some value that may be evaluated at compile time.

**XIII Resources used**

| S.No. | Name of Resource                          | Specifications |                                        | Remarks<br>(If any) |
|-------|-------------------------------------------|----------------|----------------------------------------|---------------------|
|       |                                           | Make           | Details                                |                     |
| 1.    | Computer System with broad specifications |                | Windows 10<br>i3 processor<br>2 GB RAM |                     |
| 2.    | Software                                  |                | Turbo C                                |                     |
| 3.    | Any other resource used                   |                |                                        |                     |

**XIV Results**

.....  
.....

**XV Conclusion (Actions/decisions to be taken based on the interpretation of results).**

Thus...we...used...switch...statement.....

**XVI Practical Related Questions**

Note: Below given are few sample questions for reference. Teachers must design more such questions so as to ensure the achievement of identified CO.

- Write output of following program

```
main()
{
    intch = 'a' + 'b';
    switch ( ch )
    {
```

```

case 'a':
case 'b':
printf( "\nYou entered b" );
case 'A':
printf( "\na as in ashar" );
case 'b' + 'a':
printf( "\nYou entered a and b" );

```

2. Write output of following program.

```
#include<stdio.h>
void main()
{
    int check=2;
    switch(check)
    {
        case 1: printf("D.W.Steyn");
        case 2: printf(" M.G.Johnson");
        case 3: printf(" Mohammad Asif");
        default: printf(" M.Muralidaran");
    }
}
```

3. Point out the error in the following program.

```

main( )
{
float a = 3.5 ;
switch ( a )
{
    case 0.5 :
        printf ( "\nThe art of C" ) ;
        break ;
    case 1.5 :
        printf ( "\nThe spirit of C" ) ;
        break ;
    case 2.5 :
        printf ( "\nSee through C" ) ;
        break ;
    case 3.5 :
        printf ( "\nSimply C" ) ;
}

```

## Space for Answer

2) print a student's grade by accepting present marks

```
#include <stdio.h>
#include <conio.h>
Void main()
{
    int mark;
    clrscr();
    printf("enter the marks between 0 + 100");
    scanf("%d", &mark);
    switch(mark/10)
    {
        case 10:
        case 9:
            printf("your grade is A\n"); //marks 70 - 100
            break;
        case 8:
        case 7:
            printf("your grade is B\n"); //marks 70 - 89
            break;
        case 6:
            printf("your grade is C\n"); //marks 60 - 59
            break;
        case 5:
        case 4:
            printf("your grade is D\n"); //marks 40 - 39
            break;
        default:
            printf("you are fail"); //marks less than 40
    }
    getch();
}
```

## Practical No. 8: Using switch statement

Write programs to:

1. Check whether the triangle is isosceles, equilateral, scalene or right angled triangle.
2. Display the season- Summer, Winter, Rain for given month of year [Jan., Feb, ..... , Dec.]

### I Practical Significance

Student will be able to solve geometric & logical problems using switch statements and will develop an ability to understand how multiple if-else statements can be replaced by simple switch case.

### II Relevant Program Outcomes

- **Basic knowledge:** Apply knowledge of basic mathematics, sciences and basic engineering to solve the broad-based Computer engineering problem.
- **Discipline knowledge:** Apply Information Technology knowledge to solve broad-based Information Technology related problems.
- **Experiments and practice:** Plan to perform experiments, practices and to use the results to solve Information Technology related problems.
- **Engineering tools:** Apply appropriate Information Technology related techniques/tools with an understanding of the limitations.
- **Ethics:** Apply ethical principles for commitment to professional ethics, responsibilities and norms of the practice also in the field of Computer engineering.
- **Individual and team work:** Function effectively as a leader and team member in diverse/ multidisciplinary teams.
- **Communication:** Communicate effectively in oral and written form.
- **Life-long learning:** Engage in independent and life-long learning activities in the context of technological changes in the Computer engineering field and allied industry.

### III Competency and Practical skills

This practical is expected to develop the following skills for the industry identified competency '**Develop 'C' programs to solve broad-based computer related problem**:

1. Write Algorithm for given program.
2. Draw Flow Chart for given Algorithm.
3. Develop a 'C' program using variables, constants and arithmetic expressions.
4. Debug the program.
5. Execute a simple C program using Turbo C/ gcc compiler.

### IV Relevant Course Outcome(s)

1. Develop flowchart and algorithm to solve problems logically.
2. Write simple 'C' programs using arithmetic expressions.

- V Practical Outcome**
1. Write/compile/execute simple 'C' program using Turbo 'C' and gcc compiler.
  2. Use switch statements in the program.

**VI Relevant Affective domain related Outcome(s)**

1. Follow safety practices.
2. Demonstrate working as a leader/a team member.

**VII Minimum Theoretical Background**

Basic program logic using different decision making and branching statements.

**VIII Algorithm**

Step 1 - Start

Step 2 - Declare variable -  $x, y, z$

Step 3 - Accept input from user

Step 4 - if ( $x = y \& d[y = z]$ )

print the  $\Delta$  is equilateral

if ( $x == z$ ) || ( $y == z$ ) || ( $x == y$ )

print the  $\Delta$  is isosceles if is

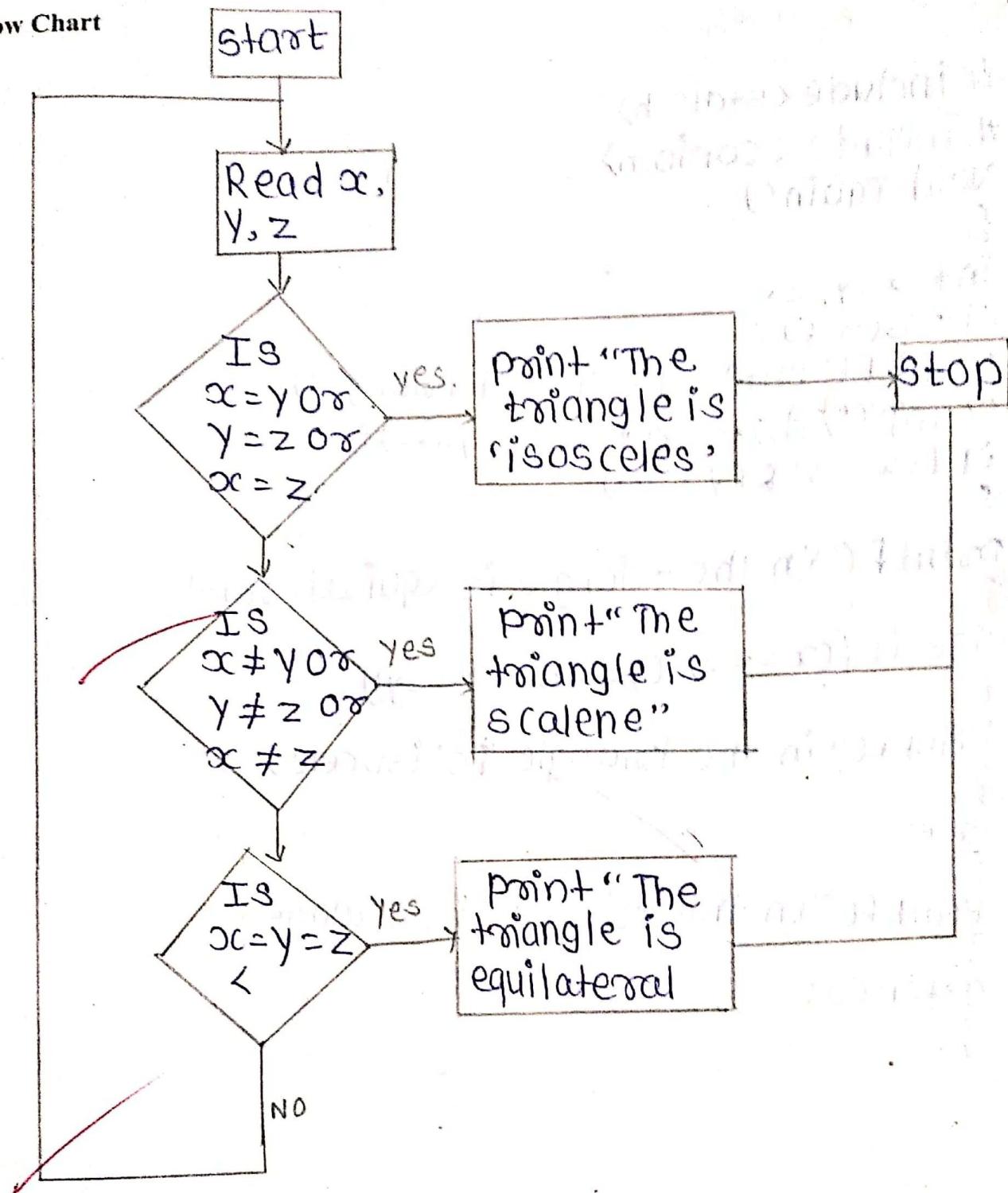
neither equilateral nor isosceles then

it

scalene  $\Delta$

Step 5 - Stop

## IX Flow Chart



**X 'C' Program Code**

```

#include <stdio.h>
#include <conio.h>
Void main()
{
    int x, y, z;
    clrscr();
    Point f("enter the side of triangle");
    Scanf ("%d,%d,%d", &x, &y, &z);
    if ((x==y) & (y==z))
    {
        Point f ("In the triangle is equilateral");
    }
    else if ((x==z) || (y==z) || (x==y))
        Point f ("In the triangle is isoscele");
    else
        Point f ("In the triangle is scalene");
    getch();
}

```

**I Resource**

| S.<br>No. |     |
|-----------|-----|
| 1.        | C   |
| 2.        | 'C' |

**I Precaut**

1. Han
2. Stric
3. Star
4. Synt
5. The
- time

**I Resourc**

| S.No. |
|-------|
| 1.    |
| 2.    |
| 3.    |

**Results****Conclus****...Thus****Practica**

Note:  
more suc

1. Ident
  - a. s
  - b. s

**XI Resources required**

| S. No. | Name of Resource | Major Specification                                     | Qty.                                 | Remarks |
|--------|------------------|---------------------------------------------------------|--------------------------------------|---------|
| 1.     | Computer System  | Any desktop or laptop computer with basic configuration | One computer system for each student |         |
| 2.     | 'C' Compiler     | Turbo C/gcc                                             | One for each computer system         |         |

**XII Precautions to be followed**

1. Handle computer system with care.
2. Strictly follow the instruction for writing, compiling and executing the program.
3. Start and Shutdown system with proper procedure.
4. Syntax for switch cases should be given properly.
5. The case conditions must be constant, or some value that may be evaluated at compile time.

**XIII Resources used**

| S.No. | Name of Resource                          | Specifications |                                       | Remarks<br>(If any) |
|-------|-------------------------------------------|----------------|---------------------------------------|---------------------|
|       |                                           | Make           | Details                               |                     |
| 1.    | Computer System with broad specifications |                | Windows 10<br>i3 processor<br>2GB RAM |                     |
| 2.    | Software                                  |                | TurboC                                |                     |
| 3.    | Any other resource used                   |                |                                       |                     |

**XIV Results**

.....  
 .....  
 .....  
 .....  
**XV Conclusion** (Actions/decisions to be taken based on the interpretation of results).  
 Thus we used switch statement.  
 .....  
 .....

**XVI Practical Related Questions**

*Note: Below given are few sample questions for reference. Teachers must design more such questions so as to ensure the achievement of identified CO.*

1. Identify valid and invalid expressions from following switch statements.
  - a.  $\text{switch}(1+2+23)$
  - b.  $\text{switch}(a+b+c)$

- c. switch( $1*2+3 \% 4$ )  
 d. switch(ab+cd)
2. Write the output of following C program, when input value 1 is entered through keyboard.

```
#include <stdio.h>
void main()
{
    int ch;
    printf("enter a value btw 1 to 2:");
    scanf("%d", &ch);
    switch (ch, ch + 1)
    {
        case 1:
            printf("1\n");
            break;
        case 2:
            printf("2");
            break;
    }
}
```

3. Write output of following program after execution.

```
#include<stdio.h>
void main()
{
    static int i;
    int j;
    for(j=0;j<=5;j+=2)
        switch(j)
        {
            case 1: i++;break;
            case 2: i+=2;
            case 4: i%2;j-=1;continue;
            default: --i;continue;
        }
    printf("%d",i);
}
```

Space for Answer

2) Output  $\rightarrow 1$

3) Output  $\rightarrow 0$

## Practical No. 9: Looping

**Write a program to:**

1. Find sum of digits of a given number.
2. Generate multiplication table up to 10 for numbers 1 to 5.

### I Practical Significance

Loops are used in programming to repeat a specific block of code. Student will be able to write programs using different loop statements.

### II Relevant Program Outcomes

- **Basic knowledge:** Apply knowledge of basic mathematics, sciences and basic engineering to solve the broad-based Computer engineering problem.
- **Discipline knowledge:** Apply Information Technology knowledge to solve broad-based Information Technology related problems.
- **Experiments and practice:** Plan to perform experiments, practices and to use the results to solve Information Technology related problems.
- **Engineering tools:** Apply appropriate Information Technology related techniques/tools with an understanding of the limitations.
- **Ethics:** Apply ethical principles for commitment to professional ethics, responsibilities and norms of the practice also in the field of Computer engineering.
- **Individual and team work:** Function effectively as a leader and team member in diverse/ multidisciplinary teams.
- **Communication:** Communicate effectively in oral and written form.
- **Life-long learning:** Engage in independent and life-long learning activities in the context of technological changes in the Computer engineering field and allied industry.

### III Competency and Practical skills

This practical is expected to develop the following skills for the industry identified competency '**Develop 'C' programs to solve broad-based computer related problem'**:

1. Write Algorithm for given program.
2. Draw Flow Chart for given Algorithm.
3. Develop a 'C' program using variables, constants and arithmetic expressions.
4. Debug the program.
5. Execute a simple C program using Turbo C/ gcc compiler.

### IV Relevant Course Outcome(s)

1. Develop flowchart and algorithm to solve problems logically.
2. Develop 'C' Programs using control structure.

### V Practical Outcome

1. Write/compile/execute simple 'C' program using Turbo 'C' and gcc compiler.
2. Ability to use For loops in the program.

**VI Relevant Affective domain related Outcome(s)**

1. Follow safety practices.
2. Practice optimal way to solve problem.

Flow Chart

**VII Minimum Theoretical Background**

Concept of decision control statements and iterative statement. Selection appropriate control or looping statements for given problem.

**VIII Algorithm**

**Step 1 : Start**

**Step 2 : Declare the variable i, j**

**Step 3 : perform nesting of loop**

```
for(i=1; i < 5; i++)
```

{

```
    for(j=1; j < 10; j++)
```

{

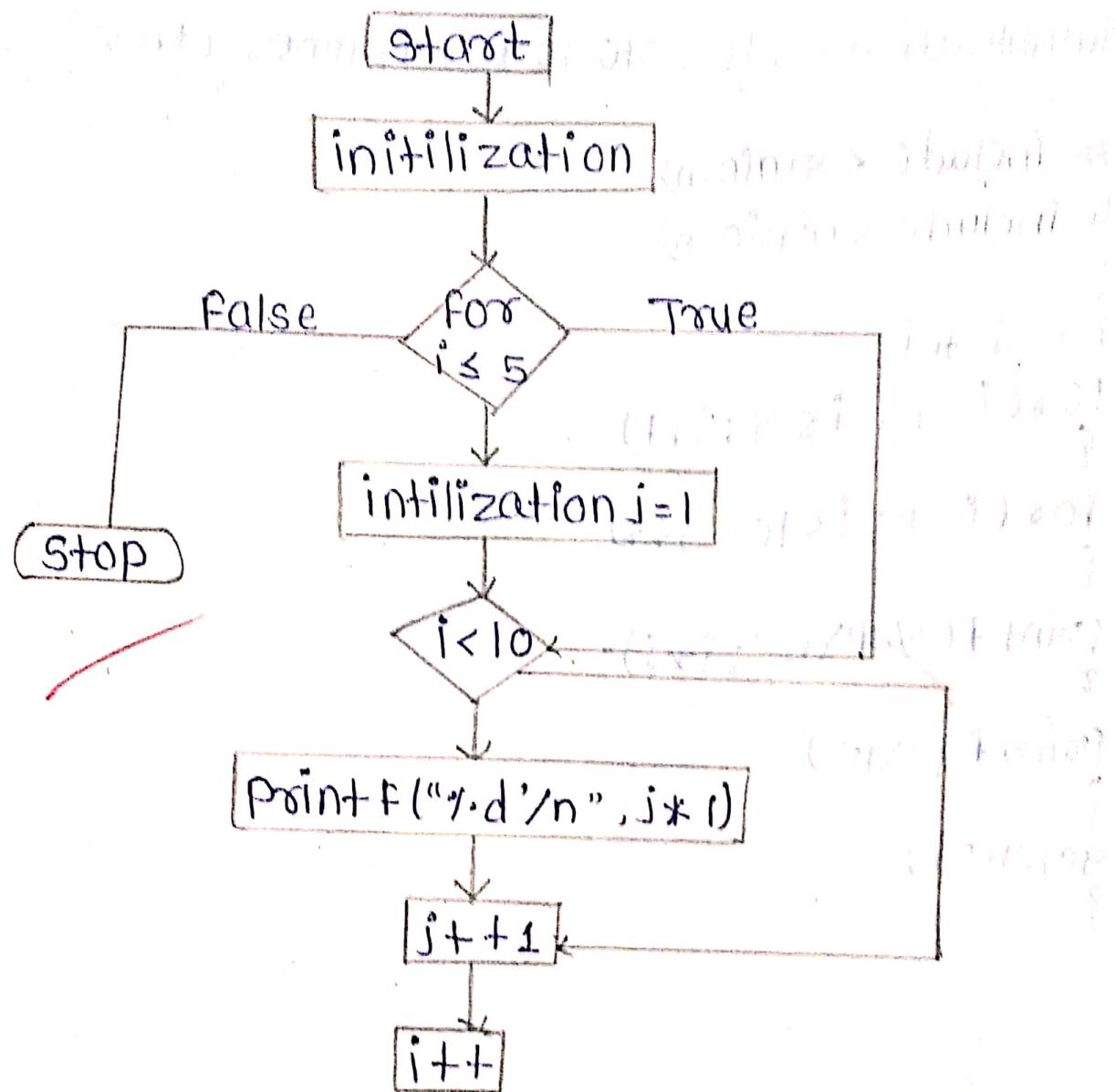
    display value of i,j.

}

**Step 4 : Stop**

(Stop)

## IX - Flow Chart



## X 'C' Program Code

Multiplication table upto 10 for number 1 to 5

```
#include <stdio.h>
#include <conio.h>
{
    int i, j;
    for(i = 1; i <= 5; i++)
    {
        for(j = 1; j <= 10; j++)
        {
            printf("%d\n", i * j);
        }
        printf("\n");
    }
    getch();
}
```

**XI Resources required**

| S.No. | Name of Resource | Major Specification                                     | Qty.                                 | Remarks |
|-------|------------------|---------------------------------------------------------|--------------------------------------|---------|
| 1.    | Computer System  | Any desktop or laptop computer with basic configuration | One computer system for each student |         |
| 2.    | 'C' Compiler     | Turbo C/gcc                                             | One for each computer system         |         |

**II Precautions to be followed**

- Handle computer system with care.
- Strictly follow the instruction for writing, compiling and executing the program.
- Start and Shutdown system with proper procedure.
- Syntax of 'for loop' should be given properly.
- Avoid infinite loop execution.

**II Resources used**

| S.No. | Name of Resource                          | Specifications |                                        | Remarks<br>(If any) |
|-------|-------------------------------------------|----------------|----------------------------------------|---------------------|
|       |                                           | Make           | Details                                |                     |
| 1.    | Computer System with broad specifications |                | Windows 10<br>i5 processor<br>2 GB RAM |                     |
| 2.    | Software                                  |                | Turboc                                 |                     |
| 3.    | Any other resource used                   |                |                                        |                     |

**V Results****V Conclusion (Actions/decisions to be taken based on the interpretation of results).**

Hence we studied loops are used in programming to repeat a specific block of code.

**VI Practical Related Questions**

Note: Below given are few sample questions for reference. Teachers must design more such questions so as to ensure the achievement of identified CO.

- Write output of following C program.

```
main()
{
int i = 0;
```

```

        for ( ; i ; )
printf( "\nHere is some mail for you" );
}

```

2. Write output of following C program after execution.

```

#include<stdio.h>
int main()
{
    int x=011,i;
    for(i=0;i<x;i+=3)
    {
        printf("Start ");
        continue;
        printf("End");
    }
    return 0;
}

```

#### Space for Answer

Q1 Sum of 5 digit number

```

....#include <stdio.h>
....#include <conio.h>
....void main();
....{
....    long int num, sum = 0;
....    clrscr();
....    printf("enter 5 digit number");
....    scanf("%d", &num);
....    while [num != 0]
....    {
....        sum = num + num % 10 / 10 / 10 / 10 / 10;
....        num = num / 10;
....    }
....    printf("total 5 digit numbers is %d", sum);
....    getch();
....}

```

**Practical No. 10 Write a program to:**

1. Find Fibonacci series for given number.
2. Produce the following output:

1  
2      3  
4      5      6  
7      8      9      10

**I Practical Significance**

Students will be able to write programs using different loop statements. This is generally used for performing a same task, a fixed number of times. Student can use nested loops using while, for, or do-while loop.

**II Relevant Program Outcomes**

- **Basic knowledge:** Apply knowledge of basic mathematics, sciences and engineering to solve the broad-based Computer engineering problem.
- **Discipline knowledge:** Apply Information Technology knowledge to solve broad-based Information Technology related problems.
- **Experiments and practice:** Plan to perform experiments, practices and to use results to solve Information Technology related problems.
- **Engineering tools:** Apply appropriate Information Technology related technical tools with an understanding of the limitations.
- **Ethics:** Apply ethical principles for commitment to professional responsibilities and norms of the practice also in the field of Computer engineering.
- **Individual and team work:** Function effectively as a leader and team member in diverse/ multidisciplinary teams.
- **Communication:** Communicate effectively in oral and written form.
- **Life-long learning:** Engage in independent and life-long learning activities in context of technological changes in the Computer engineering field and industry.

**III Competency and Practical skills**

This practical is expected to develop the following skills for the industry identified competency '*Develop 'C' programs to solve broad-based computer related problem*'

1. Write Algorithm for given program.
2. Draw Flow Chart for given Algorithm.
3. Develop a 'C' program using variables, constants and arithmetic expressions.
4. Debug the program.
5. Execute a simple C program using Turbo C/ gcc compiler.

**IV Relevant Course Outcome(s)**

1. Develop flowchart and algorithm to solve problems logically.
2. Develop 'C' programs using control structure.

**V Practical Outcome**

1. Write/compile/execute simple 'C' program using Turbo 'C' and gcc compiler.
2. Use loops and nested loops in the program.

**VI Relevant Affective domain related Outcome(s)**

1. Demonstrate working as a leader/a team member.
2. Follow ethical practices.

**VII Minimum Theoretical Background**

Concept of decision control statements and iterative statement. Selection of appropriate nested loop statements for given problem.

**III Algorithm**

Step 1 : Start

Step 2 : Declare various  $i = t_1, t_2$  next term

Step 3 : Initialize the variable  $t_1=0$   $t_2=1$  & next term = 0

Step 4 : Enter the number of term of Fibonacci series  
to be printed

Step 5 : print first two term of series

Step 6 : Use loop for the following steps next term  
 $t_1 + t_2$

$t_1 = t_2$

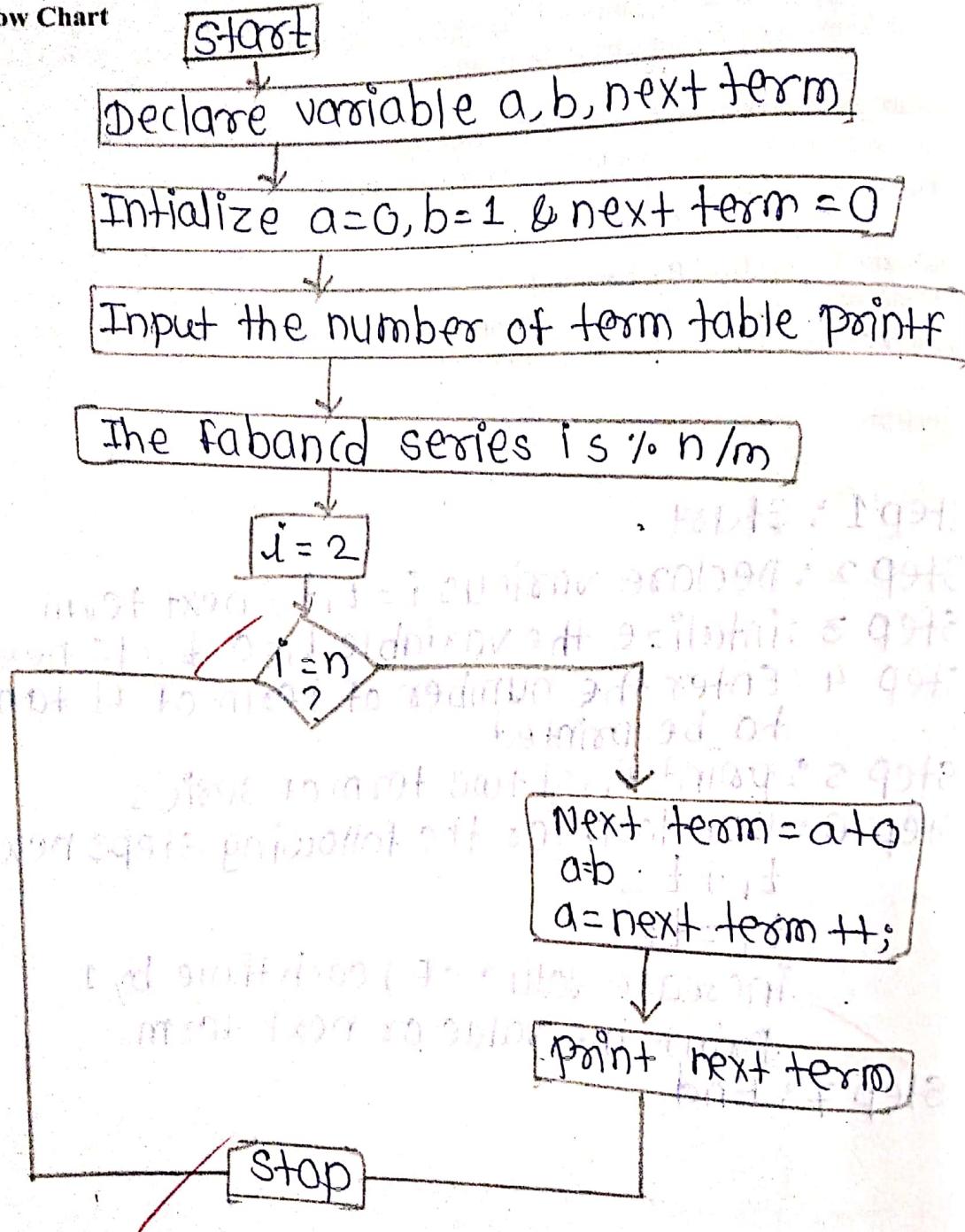
Increase value of 1 each time by 1

Print the value of next term.

Step 7 : End



## IX Flow Chart



## X 'C' Program Code

```

#include <stdio.h>
#include <conio.h>
int main()
{
    int int = 0, t2 = 1, next term
    printf("enter the number of term");
    scanf("%d", &n);
    printf("fibonacci series");
    {
        printf("%d");
        next term = t1 + t2
        t1 = t2;
        t2 = next term
    }
    return 0;
}

```

## XI Resources required

| Resources required |                  |                                                         |                                      |         |
|--------------------|------------------|---------------------------------------------------------|--------------------------------------|---------|
| S.No.              | Name of Resource | Major Specification                                     | Qty.                                 | Remarks |
| 1.                 | Computer System  | Any desktop or laptop computer with basic configuration | One computer system for each student | 100     |
| 2.                 | 'C' Compiler     | Turbo C/gcc                                             | One for each computer system         | 100     |

## XII Precautions to be followed

- Precautions to be followed**

  1. Handle computer system with care.
  2. Strictly follow the instruction for writing, compiling and executing the program.
  3. Start and Shutdown system with proper procedure.
  4. Syntax of loops should be followed properly.
  5. Avoid infinite loop execution.

### XIII      Resources used

| Resources used |                                           | Specifications |                                        | Remarks<br>(If any) |
|----------------|-------------------------------------------|----------------|----------------------------------------|---------------------|
| S.No.          | Name of Resource                          | Make           | Details                                |                     |
| 1.             | Computer System with broad specifications |                | Windows 10 is<br>processor 2 GB<br>RAM |                     |
| 2.             | Software                                  |                | Turbo C                                |                     |
| 3.             | Any other resource used                   |                |                                        |                     |

xiv Results

三

**Conclusion** (Actions/decisions to be taken based on the interpretation of results).

Thus we written a program to find Fibonacci series for given number.....

xvi

## **Practical Related Questions**

**Practical Related Questions**

Note: Below given are few sample questions for reference. Teachers must design more such questions so as to ensure the achievement of identified CO.

- Write output for following C Program.**

1. Write output for  
void main()  
{  
    int i;  
    clrscr();

```
for(i=0;i<=4;i++)
{
    printf("%d",i);
    break;
    printf("Hello");
}
printf("Programming in C");
getch();
}
```

2. Write output for following C program.

```
main()
{
    int x = 4, y = 0, z ;
    while ( x >= 0 )
    {
        x-- ;
        y++ ;
        if ( x == y )
            continue ;
        else
            printf( "\n%d %d", x, y ) ;
    }
}
```

3. Write output of following C program.

```
#include <stdio.h>
void main()
{
    int i = 4;
    do
    {
        printf("Welcome");
    } while (i<4)
}
```

Space for Answer

① Unreachable code

② Function should return a value

③ undefined symbol

④ do while statement missing

**Practical No. 11: Array****Develop a Program to:**

1. Sort list of 10 numbers.
2. Perform addition of 3x3 matrix.

**I Practical Significance**

Students will be able to write programs using arrays and store multiple variable values under one variable name with subscript and understand contiguous storage locations used in memory for storing these variables.

**II Relevant Program Outcomes**

- **Basic knowledge:** Apply knowledge of basic mathematics, sciences and basic engineering to solve the broad-based Computer engineering problem.
- **Discipline knowledge:** Apply Information Technology knowledge to solve broad-based Information Technology related problems.
- **Experiments and practice:** Plan to perform experiments, practices and to use the results to solve Information Technology related problems.
- **Engineering tools:** Apply appropriate Information Technology related techniques/tools with an understanding of the limitations.
- **Ethics:** Apply ethical principles for commitment to professional ethics, responsibilities and norms of the practice also in the field of Computer engineering.
- **Individual and team work:** Function effectively as a leader and team member in diverse/ multidisciplinary teams.
- **Communication:** Communicate effectively in oral and written form.
- **Life-long learning:** Engage in independent and life-long learning activities in the context of technological changes in the Computer engineering field and allied industry.

**III Competency and Practical skills**

This practical is expected to develop the following skills for the industry identified competency '*Develop 'C' programs to solve broad-based computer related problem*' :

1. Write Algorithm for given program.
2. Draw Flow Chart for given Algorithm.
3. Develop a 'C' program using variables, constants and arithmetic expressions.
4. Debug the program.
5. Execute a simple C program using Turbo C/ gcc compiler.

**IV Relevant Course Outcome(s)**

1. Develop flowchart and algorithm to solve problems logically.
2. Develop 'C' programs using arrays and structures.

**V Practical Outcome**

1. Write/compile/execute simple 'C' program using Turbo 'C' and gcc compiler.
2. Use single and multi-dimensional arrays in the program.

- VI Relevant Affective domain related Outcome(s)**
1. Follow safety practices.
  2. Follow ethical practices.

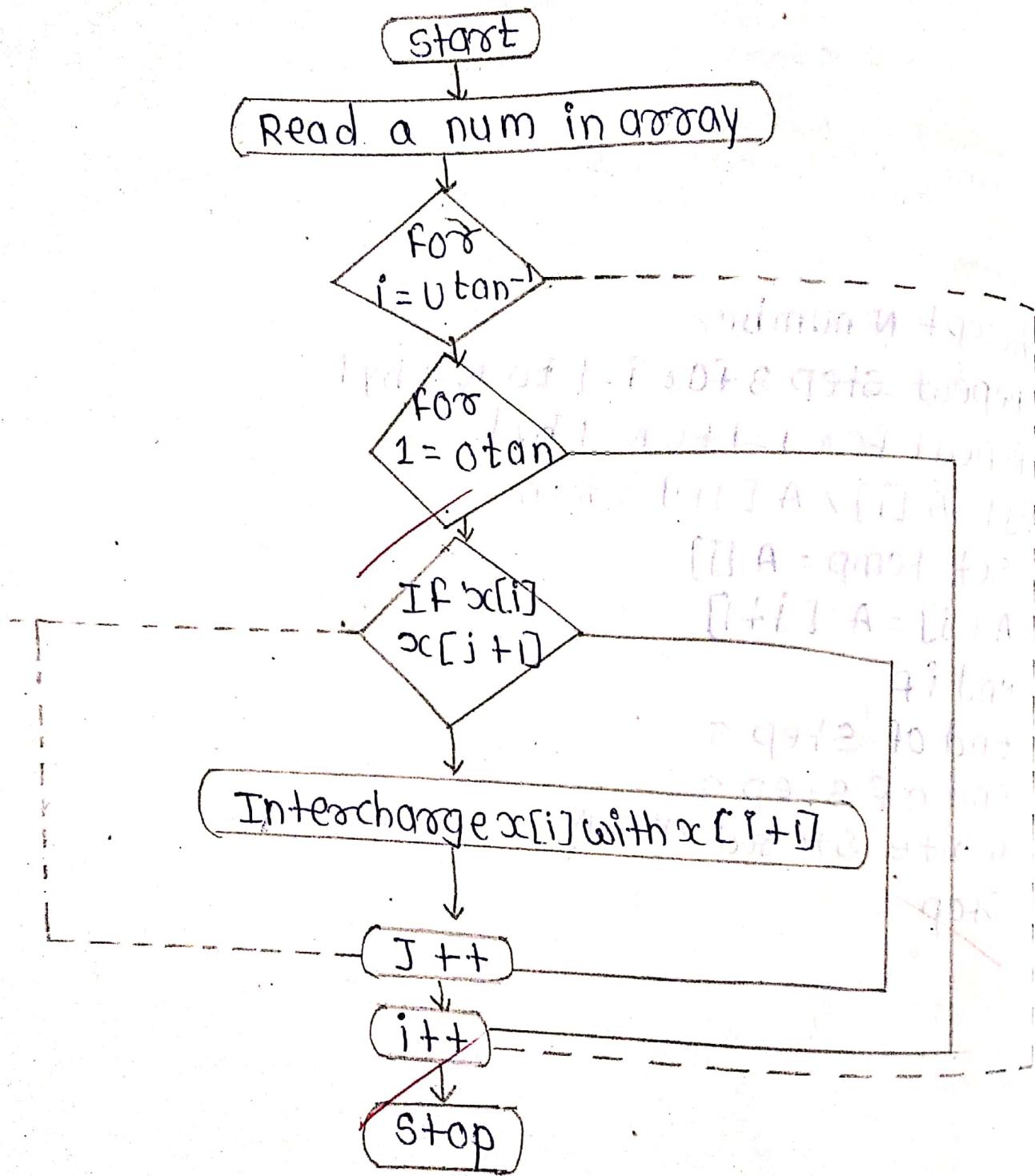
**VII Minimum Theoretical Background**

Concept of array variables and their declaration. Concept of single and multi-dimensional array.

**VIII Algorithm**

- 1) Accept N number
- 2) Repeat step 3 for  $i=1$  to  $N-1$  by 1
- 3) Repeat for  $j=1$  to  $N-1$  by 1
- 4) If  $A[i] > A[j+1]$  then
  - Set temp =  $A[i]$
  - $A[i] = A[j+1]$
  - end if
- End of Step 5
- end of step 2
- write stored array
- Stop

## IX Flow Chart



## X 'C' Program Code

```

#include <stdio.h>
#include <conio.h>
void main()
{
    int n, i, a[100], t pass;
    clrscr();
    printf("enter the size of array = 1");
    scanf("%d", &n);
    printf("\n enter array elements = ");
    for(i=0, i < n; i++)
    {
        scanf("%d", &a[i]);
    }
    printf("x n Array elements = ");
    for(i=0; i < n; i++)
    {
        printf("%d, t", a[i]);
    }
    for pass = 0; pass < n-1; pass++)
    {
        for(i=0; i < n-1; i++)
        {
            if(a[i] > a[i+1])
            {
                t = a[i];
                a[i] = a[i+1];
                a[i+1] = t;
            }
        }
        printf("inserted array element area = 1");
        for(i=0; i < n; i++)
        {
            getch();
        }
    }
}

```

C programme to sort list of 10 numbers

```

....#include <stdio.h>
....#include <conio.h>
....int main()
....{
....    int array[10], i, j, temp;
....    printf("enter 10 elements for array");
....    for (i = 0; i < 10; i++)
....    {
....        scanf("%d", &array[i]);
....    }
....    for (i = 0; i < 9; i++)
....    {
....        for (j = i + 1; j < 10; j++)
....        {
....            if (array[i] < array[j])
....            {
....                temp = array[i];
....                array[i] = array[j];
....                array[j] = temp;
....            }
....        }
....    }
....    printf("In ascending ordered array:\n");
....    for (i = 0; i < 10; i++)
....    {
....        printf("%d\t", array[i]);
....    }
....    return 1;
....}

```

## Practical No. 12: Structure

### Develop a program to:

1. Create a structure called library to hold details of a book viz.accession number, title of the book, author name, price of the book, and flag indicating whether book is issued or not. Fetch some sample data and display the same.
2. Develop and execute C Program to Add Two Distances given in kilometer-meter Using Structures.

### I Practical Significance

Student will be able to write programs using structures and use sizeof operator, typedef, Enumerated Data Type in C.

### II Relevant Program Outcomes

- **Basic knowledge:** Apply knowledge of basic mathematics, sciences and basic engineering to solve the broad-based Computer engineering problem.
- **Discipline knowledge:** Apply Information Technology knowledge to solve broad-based Information Technology related problems.
- **Experiments and practice:** Plan to perform experiments, practices and to use the results to solve Information Technology related problems.
- **Engineering tools:** Apply appropriate Information Technology related techniques/tools with an understanding of the limitations.
- **Ethics:** Apply ethical principles for commitment to professional ethics, responsibilities and norms of the practice also in the field of Computer engineering.
- **Individual and team work:** Function effectively as a leader and team member in diverse/ multidisciplinary teams.
- **Communication:** Communicate effectively in oral and written form.
- **Life-long learning:** Engage in independent and life-long learning activities in the context of technological changes in the Computer engineering field and allied industry.

### III Competency and Practical skills

This practical is expected to develop the following skills for the industry identified competency '**Develop 'C' programs to solve broad-based computer related problem**':

1. Write Algorithm for given program.
2. Draw Flow Chart for given Algorithm.
3. Develop a 'C' program using variables, constants and arithmetic expressions.
4. Debug the program.
5. Execute a simple C program using Turbo C/ gcc compiler.

### IV Relevant Course Outcome(s)

1. Develop flowchart and algorithm to solve problems logically.
2. Develop 'C' programs using arrays and structures.

V **Practical Outcome**

1. Write/compile/execute simple 'C' program using Turbo 'C' and gcc compiler.
2. Use structures in the program.

VI **Relevant Affective domain related Outcome(s)**

1. Follow safety practices.
2. Maintain tools and equipment.

VII **Minimum Theoretical Background**

Concept of structures, declaration and use. Concept of Type def and Enumerated Data Type.

VIII **Algorithm**

Step 1 : Start

Step 2 : Declare structure with its member

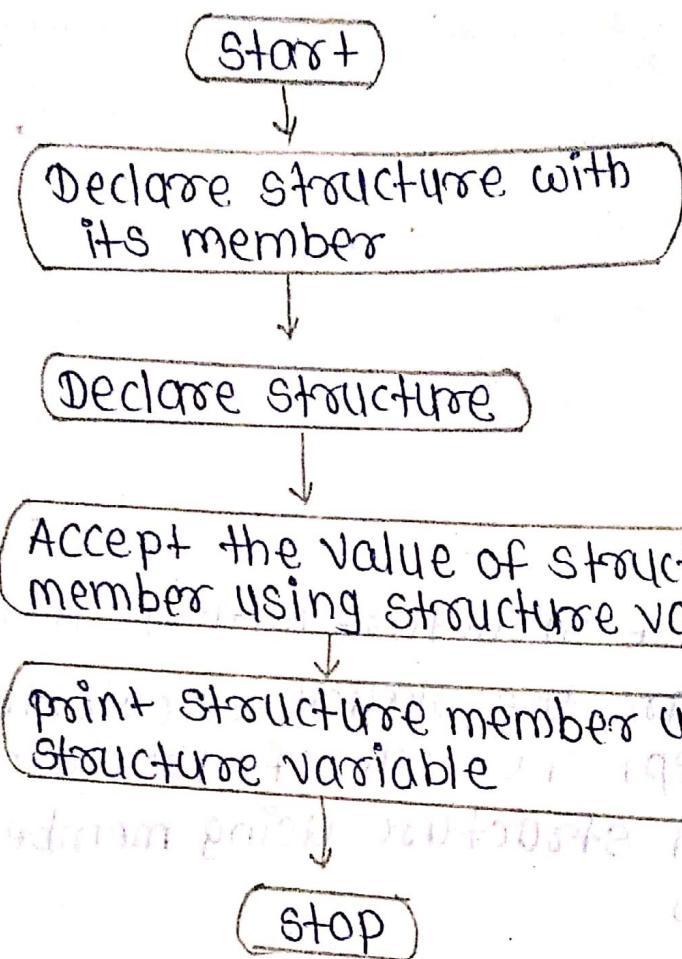
Step 3 : Declare the structure variable

Step 4 : Accept the value of structure

Step 5 : Print structure using member

Step 6 : Stop

**IX Flow Chart**



## X 'C' Program Code

```

1. #include <stdio.h>
#include <conio.h>
void main ()
{
    struct book
    char b.title [20];
    char b.author [20];
    int b.price;
};

struct book b;
printf("enter book title");
scanf("%s", b.title);
printf("enter book author");
scanf("%s", b.author);
printf("Enter book price");
scanf("%d", &b.price);
printf("book title is %s\n", b.title);
printf("book author is %s\n", b.author);
printf("book price is %d\n", &b.b.price);
getch();
}

```

Movie taken by Dinesh Bhalerao on 2017-07-22 at 10:47 AM

**XI Resources required**

| S.No. | Name of Resource | Major Specification                                     | Qty.                                 | Remarks |
|-------|------------------|---------------------------------------------------------|--------------------------------------|---------|
| 1.    | Computer System  | Any desktop or laptop computer with basic configuration | One computer system for each student |         |
| 2.    | 'C' Compiler     | Turbo C/gcc                                             | One for each computer system         |         |

**XII Precautions to be followed**

- Handle computer system with care.
- Strictly follow the instruction for writing, compiling and executing the program.
- Start and Shutdown system with proper procedure.
- Don't forget the semicolon }; in the ending line.

**XIII Resources used**

| S.No. | Name of Resource                          | Specifications |                                        | Remarks<br>(If any) |
|-------|-------------------------------------------|----------------|----------------------------------------|---------------------|
|       |                                           | Make           | Details                                |                     |
| 1.    | Computer System with broad specifications |                | Windows 10<br>i5 processor<br>2 GB RAM |                     |
| 2.    | Software                                  |                | Turbo C                                |                     |
| 3.    | Any other resource used                   |                |                                        |                     |

**XIV Results****XV Conclusion** (Actions/decisions to be taken based on the interpretation of results).

Hence we studied structure by using given implemented program successfully.

**XVI Practical Related Questions**

Note: Below given are few sample questions for reference. Teachers must design more such questions so as to ensure the achievement of identified CO.

- Write output of following C Program.

```
main()
{
    struct sample
    {
        intnum;
    }
}
```

NO output  
function ('strcpy', should).  
have a prototype

```
char m1[50];
char m2[50];
```

```
} m;
m.num = 1;
strcpy ( m.m1, "You are in the College " );
strcpy ( m.m2, "Computer Engineering" );
/* assume that the strucure is located at address 1004 */
printf ( "\n%u %u %u", &m.num, m.m1, m.m2 );
```

## 2. Point out the error in the following C code.

main()

```
{ struct
  {
    char name[25];
    char dept[10];
  };
  struct employee e = { "John", "HR" };
  printf ( "\n%s %d", e.name, e.dept );
}
```

## 3. Write output of following C Program.

#include&lt;stdio.h&gt;

```
enum year{Jan, Feb, Mar, Apr, May, Jun, Jul,
          Aug, Sep, Oct, Nov, Dec};
```

int main()

```
{
  int i;
  for (i=Jan; i<=Dec; i++)
    printf("%d ", i);
  return 0;
}
```

Space for Answer

2&gt;#include &lt;stdio.h&gt;

#include &lt;conio.h&gt;

Struct hi

{

int k1, k2;

? kilo;

```

main()
{
    printf("enter the first distance in parts of kilometer & meters respectively \n");
    scanf("%d", &kil.k1);
    scanf("%d", &kil.m1);
    printf("enter the second distance in parts of kilometer & meters respectively \n");
    scanf("%d", &kil.k2);
    scanf("%d", &kil.m2);
    printf("select the operation 1.Addition\n 2.subtract\n 3.Read\n 4.Display\n 5.Exit\n");
    scanf("%d", &a);
    if(a==1)
    {
        printf("Addition %d kilometer & %d meter, kil.k1+kil.k2, kil.m1+kil.m2");
    }
    else if(a==2)
    {
        printf("Subtraction %d kilometer & %d meter, %d kilometer & %d meters", kil.k1-kil.k2, kil.m1-kil.m2);
        printf("Distance entered by you: %d kilometer & %d meters", kil.k2, kil.m2);
    }
    else if(a==5)
    {
        exit(1);
    }
    else
    {
        printf("not valid operation");
    }
}

```

## Practical No. 13: Library Functions

### **Develop Program to demonstrate:**

1. Use of all String handling functions.
2. Use of few Mathematical functions.
3. Use of few other miscellaneous functions.

### **I Practical Significance**

Student will be able to define and declare user defined functions and library functions and use different string handling functions, mathematical functions as well as miscellaneous function. Student will develop an ability to compare strings, copy strings, generate random numbers, sine, cosine and log values.

### **II Relevant Program Outcomes**

- **Basic knowledge:** Apply knowledge of basic mathematics, sciences and engineering to solve the broad-based Computer engineering problem.
- **Discipline knowledge:** Apply Information Technology knowledge to solve broad-based Information Technology related problems.
- **Experiments and practice:** Plan to perform experiments, practices and to use the results to solve Information Technology related problems.
- **Engineering tools:** Apply appropriate Information Technology related techniques/tools with an understanding of the limitations.
- **Ethics:** Apply ethical principles for commitment to professional ethics, responsibilities and norms of the practice also in the field of Computer engineering.
- **Individual and team work:** Function effectively as a leader and team member in diverse/ multidisciplinary teams.
- **Communication:** Communicate effectively in oral and written form.
- **Life-long learning:** Engage in independent and life-long learning activities in the context of technological changes in the Computer engineering field and allied industry.

### **III Competency and Practical skills**

This practical is expected to develop the following skills for the industry identified competency '*Develop 'C' programs to solve broad-based computer related problem*':

1. Write Algorithm for given program.
2. Draw Flow Chart for given Algorithm.
3. Develop a 'C' program using variables, constants and arithmetic expressions.
4. Debug the program.
5. Execute a simple C program using Turbo C/ gcc compiler.

### **IV Relevant Course Outcome(s)**

1. Develop flowchart and algorithm to solve problems logically.
2. Develop/Use functions in C programs for modular programming approach.

V **Practical Outcome**

1. Write/compile/execute simple 'C' program using Turbo 'C' and gcc compiler.
2. Use Library and user defined functions in the program.

VI **Relevant Affective domain related Outcome(s)**

1. Maintain tools and equipment.
2. Follow ethical practices.

VII **Minimum Theoretical Background**

Concept and need of functions. Concept of Library functions like Math functions, String handling functions and other miscellaneous functions. Declaration and use of User defined functions.

VIII **Algorithm**

Write a program that accepts your name & state its length

Algorithm:

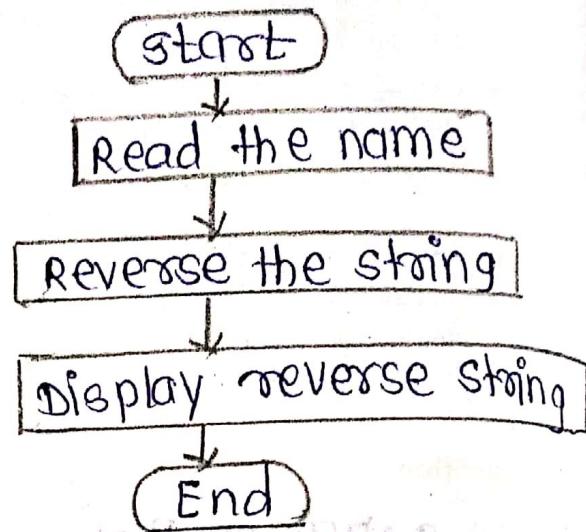
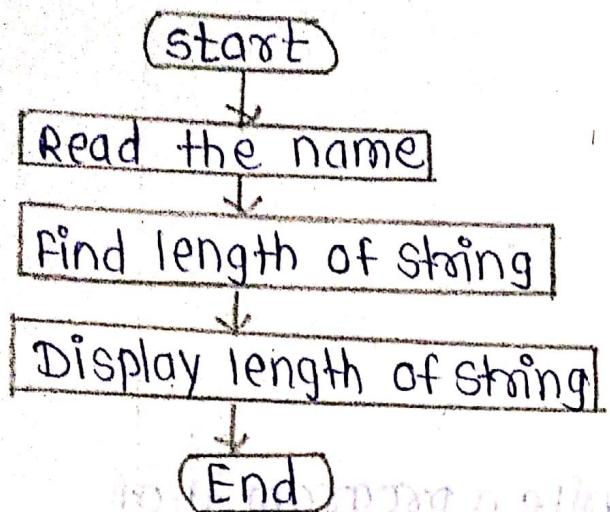
- 1) Start
- 2) Read the name
- 3) Include all required head on files
- 4) Read the name
- 5) Find length of string
- 6) Display length of string
- 7) End

Write a program that accepts your name & prints it in reverse order

Algorithm:

- 1) Start
- 2) Include all required head on files
- 3) Read the name
- 4) Reverse the string
- 5) Display reverse the string
- 6) End

## IX Flow Chart



## X 'C' Program Code

```
#include <stdio.h>
#include <string.h>
void main ()
{
    char a [20];
    int i;
    gets (a);
    printf ("%d", i);
}
```

```
#include <stdio.h>
#include <string.h>
void main ()
{
    char a (30);
    char b [30];
    clrscr ();
    printf ("enter your name");
    getch (a);
    b = strrev (a);
    printf ("Reverse of string is %s", b);
    getch ();
}
```

**XI Resources required**

| S.No. | Name of Resource | Major Specification                                     | Qty.                                 | Remarks |
|-------|------------------|---------------------------------------------------------|--------------------------------------|---------|
| 1.    | Computer System  | Any desktop or laptop computer with basic configuration | One computer system for each student |         |
| 2.    | 'C' Compiler     | Turbo C/gcc                                             | One for each computer system         |         |

**XII Precautions to be followed**

- Handle computer system with care.
- Strictly follow the instruction for writing, compiling and executing the program.
- Start and Shutdown system with proper procedure.
- Don't forget to include header file related to particular function.

**XIII Resources used**

| S.No. | Name of Resource                          | Specifications |                                     | Remarks<br>(If any) |
|-------|-------------------------------------------|----------------|-------------------------------------|---------------------|
|       |                                           | Make           | Details                             |                     |
| 1.    | Computer System with broad specifications |                | Windows 10 15 processor or 2 GB RAM |                     |
| 2.    | Software                                  |                | Turbo C                             |                     |
| 3.    | Any other resource used                   |                |                                     |                     |

**XIV Results**

Hence we studied library function by using given implemented program successfully.

**XV Conclusion** (Actions/decisions to be taken based on the interpretation of results).**XVI Practical Related Questions**

Note: Below given are few sample questions for reference. Teachers must design more such questions so as to ensure the achievement of identified CO.

1. Write output of following C Program.

```
#include <stdio.h>
#include <math.h>
int main()
{
```

i) Undeclared symbol 'int'  
ii) Undefined symbol '0'  
iii) function call missing.

```

int i = 10;
printf("%f\n", log10(i));
return 0;
}

```

2. Find the errors in the following C program:

```

main()
{
    charstr[] = "Good Morning";
    if (strstr(str, " Welcome") == 0)
        printf("\n substring Found");
}

```

- i) Undefined symbol 'characterstic'
- ii) expression syntax
- iii) function 'strstr' should have a prototype
- iv) Undefined Symbol 'str'

#### Space for Answer

```

#include <stdio.h>
#include <math.h>
int main(void)
{
    long result; long xc = -12345678;
    int num = -1234;
    printf(" %d is absolute value of integer %d \n",
           abs(num));
    result = !abs(xc);
    printf(" %d is absolute value of long %d \n",
           result);
    double number = 625.54;
    double down = up;
    down = floor(number);
    up = ceil(number);
    printf("original number %.2f \n", number);
    printf("number rounded down %.5.2f \n", down);
    printf("number rounded up %.5.2f \n", up);
    printf("square root of the number %.5.2f is",
           sqrt(down));
    return;
}

```

## Practical No.14: User Defined Functions

**Develop a Program to:**

1. Create a function to find GCD of given number. Call this function in a program.
2. Find Factorial of given number using recursion.

### I Practical Significance

Student will be able to define and declare user define functions and can understand the scope of variables. Student will be able to use parameters using call by value and call by reference in C program and use recursive functions.

### II Relevant Program Outcomes (POs)

- **Basic knowledge:** Apply knowledge of basic mathematics, sciences and basic engineering to solve the broad-based Computer engineering problem.
- **Discipline knowledge:** Apply Information Technology knowledge to solve broad-based Information Technology related problems.
- **Experiments and practice:** Plan to perform experiments, practices and to use the results to solve Information Technology related problems.
- **Engineering tools:** Apply appropriate Information Technology related techniques/tools with an understanding of the limitations.
- **Ethics:** Apply ethical principles for commitment to professional ethics, responsibilities and norms of the practice also in the field of Computer engineering.
- **Individual and team work:** Function effectively as a leader and team member in diverse/ multidisciplinary teams.
- **Communication:** Communicate effectively in oral and written form.
- **Life-long learning:** Engage in independent and life-long learning activities in the context of technological changes in the Computer engineering field and allied industry.

### III Competency and Practical skills

This practical is expected to develop the following skills for the industry identified competency '*Develop 'C' programs to solve broad-based computer related problem*:

1. Write Algorithm for given program.
2. Draw Flow Chart for given Algorithm.
3. Develop a 'C' program using variables, constants and arithmetic expressions.
4. Debug the program.
5. Execute a simple C program using Turbo C/ gec compiler.

### IV Relevant Course Outcome(s)

1. Develop flowchart and algorithm to solve problems logically.
2. Develop/Use functions in C programs for modular programming approach.

**V Practical Outcome**

1. Write/compile/execute simple 'C' program using Turbo 'C' and gcc compiler.
2. Write programs using User Defined Functions.

**VI Relevant Affective domain related Outcome(s)**

1. Demonstrate working as a leader/a team member.
2. Follow ethical practices.

**VII Minimum Theoretical Background**

Concept of function declaration, function definition and function call.  
Understanding of user defined functions and recursive functions.

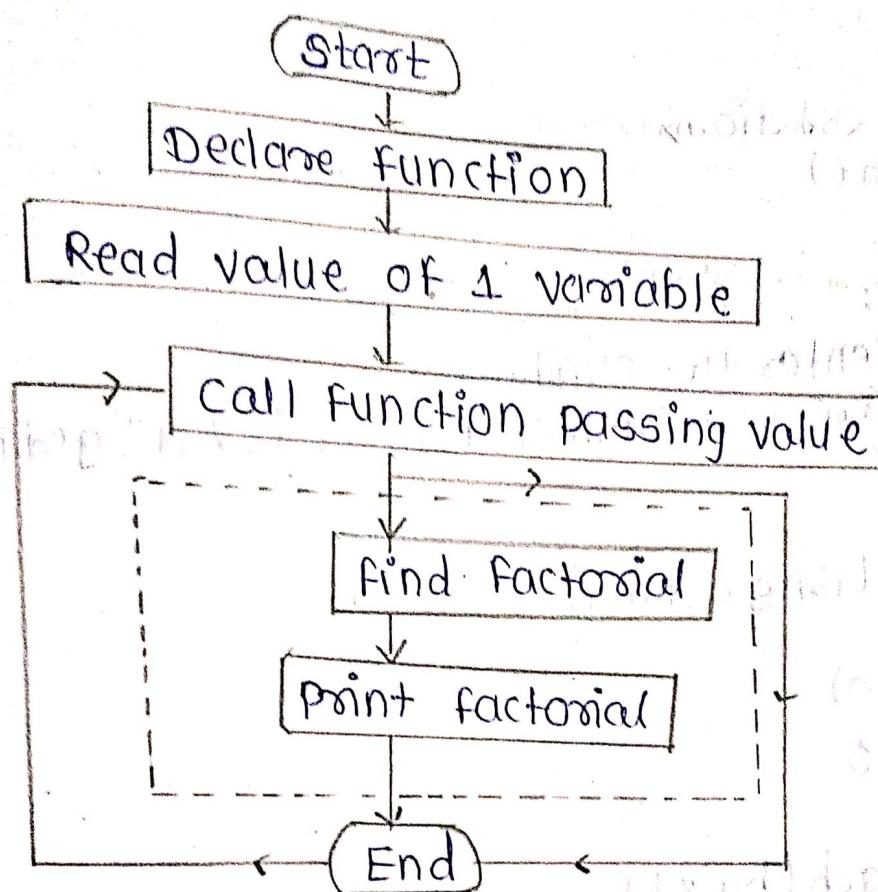
**VIII Algorithm**

i]

- i) Start
- ii) include all required header files
- iii) declare a function
- iv) input one number
- v) call function passing value
  - 1) calculate the factorial
  - 2) Return the factorial
- vi) End



## IX Flow Chart



## X 'C' Program Code

```
#include <stdio.h>
int main()
{
    int a, b;
    printf("enter two numbers");
    scanf("%d %d", &a, &b);
    return 0
}

int gcd (long a, long b)
{
    if (b == 0)
        return a;
    else
        return gcd(b % a, b);
```

## XI Resources required

| S.No. | Name of Resource | Major Specification                                     | Qty.                                 | Remarks |
|-------|------------------|---------------------------------------------------------|--------------------------------------|---------|
| 1.    | Computer System  | Any desktop or laptop computer with basic configuration | One computer system for each student |         |
| 2.    | 'C' Compiler     | Turbo C/gcc                                             | One for each computer system         |         |

## XII Precautions to be followed

- Handle computer system with care.
- Strictly follow the instruction for writing, compiling and executing the program.
- Start and Shutdown system with proper procedure.
- Function name should start with alphabet and may consist of alphanumeric characters.
- If there are two words in function name it should be attached with \_ or -. Example: emp\_salary( ), stud-data( ).
- User define function name should not be same as any built-in function.

## XIII Resources used

| S.No. | Name of Resource                          | Specifications |                                                  | Remarks<br>(If any) |
|-------|-------------------------------------------|----------------|--------------------------------------------------|---------------------|
|       |                                           | Make           | Details                                          |                     |
| 1.    | Computer System with broad specifications |                | Windows 10 i3<br>PROCESSOR 2GB<br>RAM<br>TURBO C |                     |
| 2.    | Software                                  |                |                                                  |                     |
| 3.    | Any other resource used                   |                |                                                  |                     |

## XIV Results

.....

.....

.....

## XV Conclusion (Actions/decisions to be taken based on the interpretation of results).

Hence we studied user defined functions by using give implemented program successfully.

## XVI Practical Related Questions

Note: Below given are few sample questions for reference. Teachers must design more such questions so as to ensure the achievement of identified CO.

- Write output of following C program.  
main()

```

    {
        int i = 45, c ;
        c = check( i );
        printf( "\n%d", c );
    }
    check( int ch )
    {
        if ( ch >= 45 )
            return ( 100 );
        else
            return ( 10 * 10 );
    }

```

2. Find error in following function, if any.

```

sqr( a );
int a ;
{
    return ( a * a );
}

```

3. Write output of following C program.

```

#include<stdio.h>
int fun(int count)
{
    printf("%d\n", count);
    if(count < 3)
    {
        fun(fun(++count));
    }
    return count;
}
int main()
{
    fun(1);
    return 0;
}

```

Space for Answer

```

#include <stdio.h>
int find_factorial(int);
int main()
{
    int numfact,

```

```
printf("...enter any integer number:");
scanf("%d") & num;
```

i) calling own user defined function  
 $\text{fact} = \text{find-factorial}(\text{num});$

ii) displaying factorial of input number  
 $\text{printf}("\n factorial of %d is : %d", \text{num}, \text{fact});$   
 $\text{return } 0;$

int find-factorial (int n)
{
 i. (Factorial of 0 is)
 if ( $n == 0$ )
 return (1);

ii) function calling if self : recursion  
 $\text{return } (\text{n} * \text{find-factorial}(\text{n}-1));$

iii) undefined symbol 'inti'  
 ii) undefined symbol 'c'  
 iii) function check should have a prototype

iv) Declaration was excepted constant

v) i) Undetermined string or charstr constant  
 ii) Function call missing  
 iii) Function should return to value  
 iv) Function should return a value

## Practical No.15: Pointers

Develop a Program to Print values of variables and their addresses.

### I Practical Significance

Student will be able to understand concept of pointer variables and how to access their addresses. He/she will be able to declare, initialize and access pointers. After completion of this practical student will be able to use pointers to get the memory address of variable.

### II Relevant Program Outcomes (POs)

- **Basic knowledge:** Apply knowledge of basic mathematics, sciences and basic engineering to solve the broad-based Computer engineering problem.
- **Discipline knowledge:** Apply Information Technology knowledge to solve broad-based Information Technology related problems.
- **Experiments and practice:** Plan to perform experiments, practices and to use the results to solve Information Technology related problems.
- **Engineering tools:** Apply appropriate Information Technology related techniques/tools with an understanding of the limitations.
- **Ethics:** Apply ethical principles for commitment to professional ethics, responsibilities and norms of the practice also in the field of Computer engineering.
- **Individual and team work:** Function effectively as a leader and team member in diverse/ multidisciplinary teams.
- **Communication:** Communicate effectively in oral and written form.
- **Life-long learning:** Engage in independent and life-long learning activities in the context of technological changes in the Computer engineering field and allied industry.

### III Competency and Practical skills

This practical is expected to develop the following skills for the industry identified competency '*'Develop 'C' programs to solve broad-based computer related problem'*:

1. Write Algorithm for given program.
2. Draw Flow Chart for given Algorithm.
3. Develop a 'C' program using variables, constants and arithmetic expressions.
4. Debug the program.
5. Execute a simple C program using Turbo C/ gcc compiler.

### IV Relevant Course Outcome(s)

1. Develop flowchart and algorithm to solve problems logically.
2. Develop 'C' programs using pointers.

### V Practical Outcome

1. Write/compile/execute simple 'C' program using Turbo 'C' and gcc compiler.
2. Write programs using pointers.

**VI Relevant Affective domain related Outcome(s)**

1. Follow safety practices.
2. Demonstrate working as a leader/a team member.

**VII Minimum Theoretical Background**

Concept of control structures. Understanding of variable storage and type. Declaration of pointers and assigning values to pointer variables.

**VIII Algorithm**

Step 1 : Start

Step 2 : Declaration of variable  $n = 50$ ;

Step 3 : Declaration of pointer \* p

Step 4 : Initialization of point

Step 5 : Print the value of variable

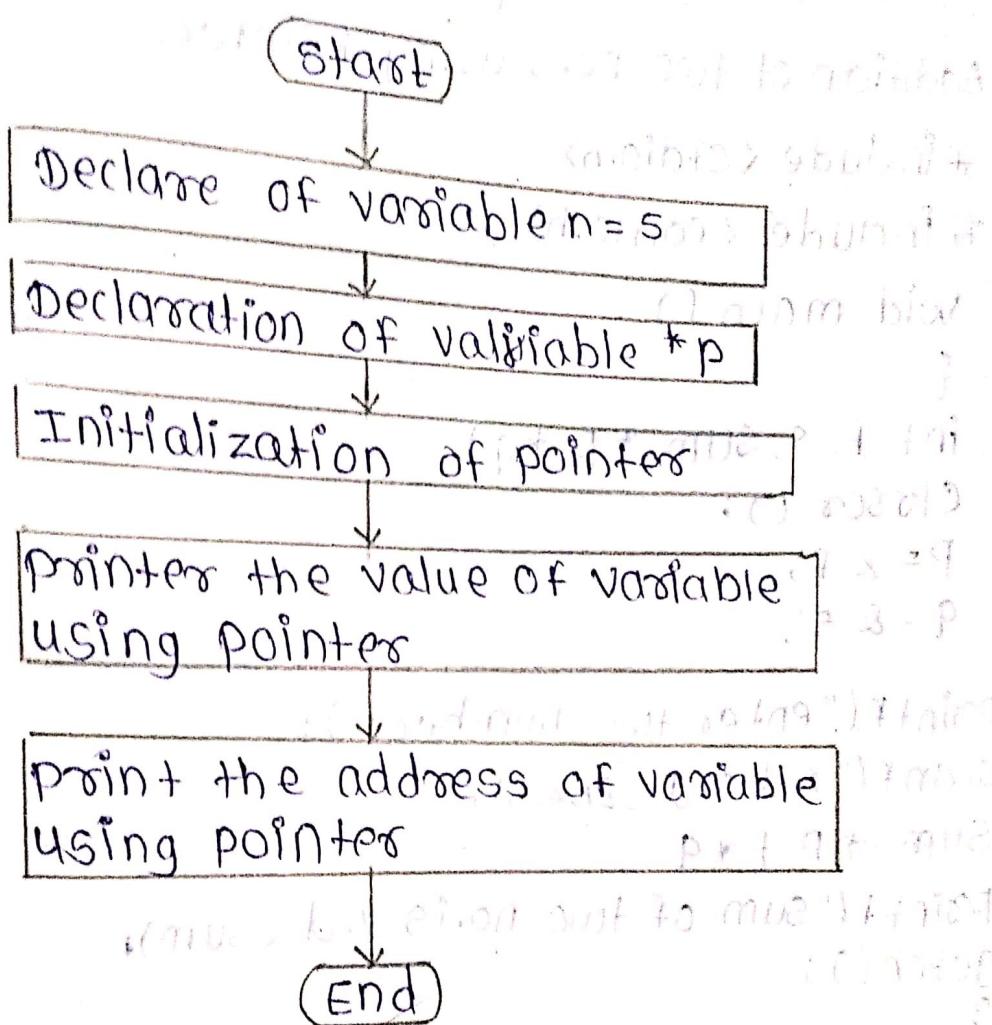
Step 6 : print the value of variable \* pointer

Step 7 : Print the address of variable using  
\* pointer

Step 8 : Stop



## IX Flowchart



## X. 'C' Program Code

Addition of two no.s using pointers

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

```
Void main()
```

```
{
```

```
int f, s, sum, *p, *q)
```

```
closecr();
```

```
p = &f;
```

```
q = &s;
```

```
printf("enter two numbers");
```

```
scanf("%d,%d", &f, &s);
```

```
sum = *p + *q
```

```
printf("sum of two no. is %d ; sum);
```

```
getch();
```

```
}
```

## XI Resources required

| S.No. | Name of Resource | Major Specification                                     | Qty.                                 | Remarks |
|-------|------------------|---------------------------------------------------------|--------------------------------------|---------|
| 1.    | Computer System  | Any desktop or laptop computer with basic configuration | One computer system for each student |         |
| 2.    | 'C' Compiler     | Turbo C/geo                                             | One for each computer system         |         |

## XII Precautions to be followed

1. Handle computer system with care.
2. Strictly follow the instruction for writing, compiling and executing the program.
3. Start and Shutdown system with proper procedure.
4. Should take care to define and declare pointers.

## XIII Resources used

| S.No. | Name of Resource                          | Make | Specifications                                       | Remarks<br>(If any) |
|-------|-------------------------------------------|------|------------------------------------------------------|---------------------|
|       |                                           |      | Details                                              |                     |
| 1.    | Computer System with broad specifications |      | Windows 10 16<br>Processor 2.6 GHz<br>RAM<br>Turbo C |                     |
| 2.    | Software                                  |      |                                                      |                     |
| 3.    | Any other resource used                   |      |                                                      |                     |

## XIV Results

.....  
.....

## XV Conclusion (Actions/decisions to be taken based on the interpretation of results).

Hence, we studied, to developing program on elements stored in given array using pointers, implemented and program successfully.

## XVI Practical Related Questions

Note: Below given are few sample questions for reference. Teachers must design more such questions so as to ensure the achievement of identified CO.

1. Assume that float takes 4 bytes, predict the output of following program.  

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

```
int main()
{
    floatarr[5] = {12.5, 10.0, 13.5, 90.5, 0.5};
```

```
float *ptr1 = &arr[0];
float *ptr2 = ptr1 + 3;
printf("%f", *ptr2);
printf("%d", ptr2 - ptr1);
return 0;
}
```

undefined symbol float are  
expression str syntax

undefined symbol are  
Undefine.

2. Write output of following C program.

```
#include <stdio.h>
void main()
{
    char a[10][5] = {"hi", "hello", "fellows"};
    printf("%d", sizeof(a[1]));
}
```

To many Initialization

Space for Answer

## Practical No.16: Develop a Program to find sum of all elements stored in given array using pointers.

### I Practical Significance

Student will be able to understand concept of pointers. He/she will be able to declare, initialize and access pointers and perform various arithmetic operations on pointers, handling arrays using pointers. After the completion of this practical student will be able to handle functions and structures using pointers.

### II Relevant Program Outcomes (POs)

- **Basic knowledge:** Apply knowledge of basic mathematics, sciences and basic engineering to solve the broad-based Computer engineering problem.
- **Discipline knowledge:** Apply Information Technology knowledge to solve broad-based Information Technology related problems.
- **Experiments and practice:** Plan to perform experiments, practices and to use the results to solve Information Technology related problems.
- **Engineering tools:** Apply appropriate Information Technology related techniques/tools with an understanding of the limitations.
- **Ethics:** Apply ethical principles for commitment to professional ethics, responsibilities and norms of the practice also in the field of Computer engineering.
- **Individual and team work:** Function effectively as a leader and team member in diverse/ multidisciplinary teams.
- **Communication:** Communicate effectively in oral and written form.
- **Life-long learning:** Engage in independent and life-long learning activities in the context of technological changes in the Computer engineering field and allied industry.

### III Competency and Practical skills

This practical is expected to develop the following skills for the industry identified competency '***Develop 'C' programs to solve broad-based computer related problem***:

1. Write Algorithm for given program.
2. Draw Flow Chart for given Algorithm.
3. Develop a 'C' program using variables, constants and arithmetic expressions.
4. Debug the program.
5. Execute a simple C program using Turbo C/ gcc compiler.

### IV Relevant Course Outcome(s)

1. Develop flowchart and algorithm to solve problems logically.
2. Develop 'C' programs using pointers.

### V Practical Outcome

1. Write/compile/execute simple 'C' program using Turbo 'C' and gcc compiler.
2. Write programs using array of pointers.

- VI Relevant Affective domain related Outcome(s)
1. Demonstrate working as a leader/a team member.
  2. Maintain tools and equipment.

VII Minimum Theoretical Background

Concept of control structures. Understanding of variable storage and type. Declaration of pointers and assigning values to pointer variables. accessing arrays using pointers. Pointer Arithmetic.

VIII Algorithm

Step 1 : Start

Step 2 : Declaration of v en s ; sum

Step 3 : Pointer dec \* pt

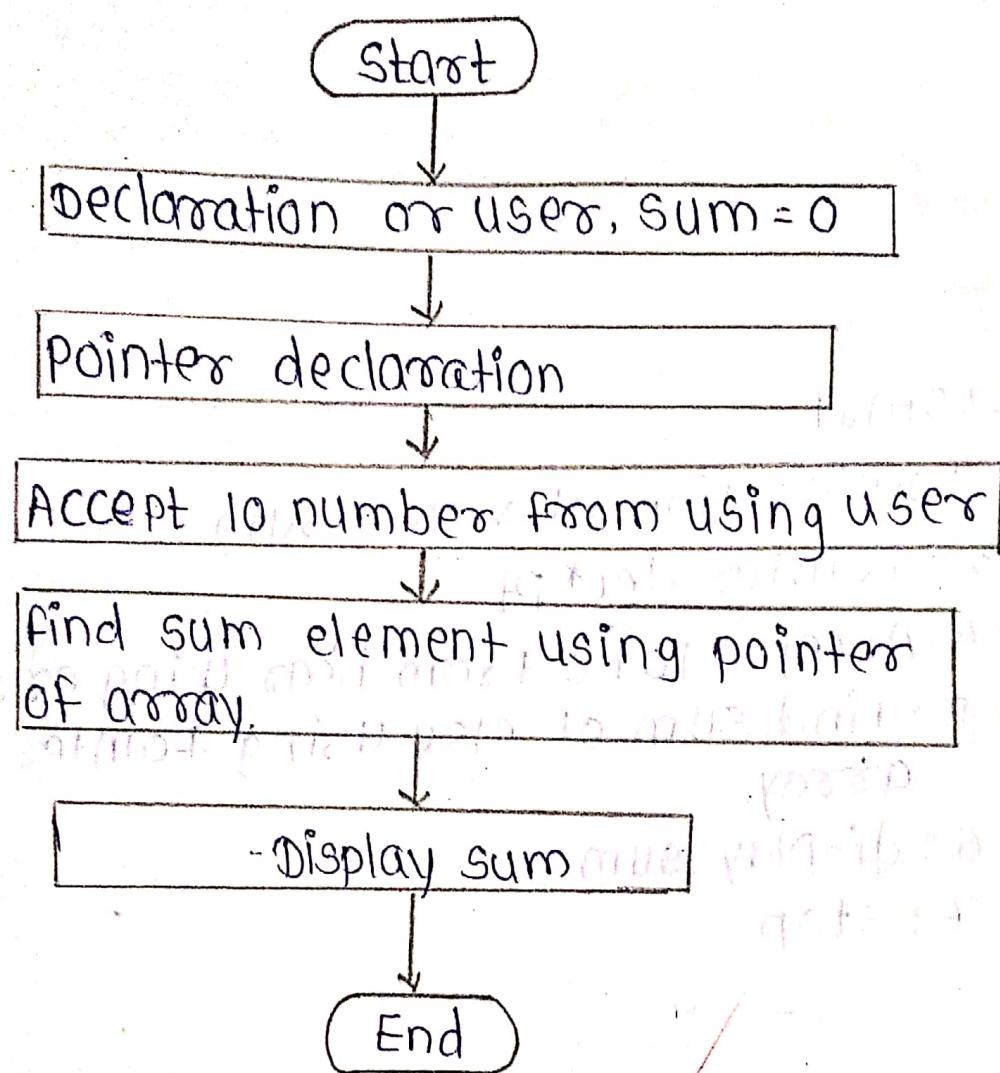
Step 4 : Accept 10 no. from hear Using array

Step 5 : Find sum of else using pointer of array.

Step 6 : display sum

Step 7 : Stop

## IX Flowchart



```

#include <stdio.h>
#include <conio.h>
void main()
{
    int num Array [10];
    int sum = 0
    int * ptn;
    printf("In enter 10 element");
    for(i=0 ; i<10 ; i++)
        scanf("%d", &num array [i]);
    ptn = num array; /* a=&a [0] */
    for(i=0 ; i<10 ; i++)
    {
        sum = sum + *ptn;
        ptn++;
    }
    printf("The sum of array element c/d", sum);
}

```

**XI Resources required**

| S.No. | Name of Resource | Major Specification                                     | Qty.                                 | Remarks |
|-------|------------------|---------------------------------------------------------|--------------------------------------|---------|
| 1.    | Computer System  | Any desktop or laptop computer with basic configuration | One computer system for each student |         |
| 2.    | 'C' Compiler     | Turbo C/gcc                                             | One for each computer system         |         |

**XII Precautions to be followed**

- Handle computer system with care.
- Strictly follow the instruction for writing, compiling and executing the program.
- Start and Shutdown system with proper procedure.
- Array name should start with alphabet. Array name may consist of alphanumeric characters.
- If there are two words in Array name it should be attached with \_ or -. Example: Array\_one[ ], Array-two[ ].
- Array name should not be keyword same as any built-in function name.

**XIII Resources used**

| S.No. | Name of Resource                          | Specifications |                                        | Remarks<br>(If any) |
|-------|-------------------------------------------|----------------|----------------------------------------|---------------------|
|       |                                           | Make           | Details                                |                     |
| 1.    | Computer System with broad specifications |                | windows 10, i5<br>PROCESSOR 2GB<br>RAM |                     |
| 2.    | Software                                  |                | TURBO C                                |                     |
| 3.    | Any other resource used                   |                |                                        |                     |

**XIV Results**

.....  
.....  
.....  
.....

**XV Conclusion** (Actions/decisions to be taken based on the interpretation of results).

Hence we studied develop a program to find sum of all elements to add in given array using pointers by using given implemented program successfully.

**XVI Practical Related Questions**

Note: Below given are few sample questions for reference. Teachers must design more such questions so as to ensure the achievement of identified CO.

- Write output of following program.  

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

```

void example( int[] );
int main()
{
    intary[4] = {1, 2, 3, 4};
    example(ary);
    printf("%d ", ary[0]);
}
void example(int p[4])
{
    int i = 10;
    p = &i;
    printf("%d ", p[0]);
}

```

2. Write output of following program.

```

#include <stdio.h>
int main()
{
    intary[4] = {1, 2, 3, 4};
    int *p = ary + 3;
    printf("%d %d\n", p[-2], ary[*p]);
}

```

#### Space for Answer

- 1) i) Undefined symbol 'intary'  
 ii) Expression syntax  
 iii) Undefined symbol 'ary'  
 iv) Function call missing

- 2) i) Undefined symbol 'intary'  
 ii) Expression syntax  
 iii) Undefined symbol 'ary'  
 iv) Function should return a value