

38

OUTPUT :

* * * * * * * Demonstration of datatype * * * * * *

Roll no. of the student :

78

Name of the student :

Preet

Mobile no. of the student :

7878787878

Percentage of the student :

80

Grade of student :

A

Address of the student :

Mumbai

Roll no. of student : 78

Name of student : Preet

mobile no. of student : 7878787878

Percentage of the student : 80

Grade of student : A

Add of student : Mumbai

Aim : Programs to understand the basis of datatypes & I/O

Source code :

```
#include <stdio.h>
#include <conio.h>
void main()
{
    int roll;
    char name[20];
    long int mob;
    float per;
    char grade;
    char add[50];
    clrscr();
    printf("***** Demonstration of datatype *****\n");
    printf("Roll no. of the student : \n");
    scanf("%d", &roll);
    printf("enter the name of the student : \n");
    scanf(gets(name));
    printf("Mobile no. of student : \n");
    scanf("%10s", &name);
    printf("Percentage of student : \n");
    scanf("%f", &per);
```

```
printf("Grade of student: %n");
scanf("%c", &grade);
printf("Address of the student: %n");
scanf("%c", &add);
printf("Roll no. of student: %d \n", roll);
printf("Name of student: %s \n", name);
printf("Mobile no. of student: %10s \n", mob);
printf("\n Percentage of student: %f \n", Per);
printf("\n Grade of student: %c \n", grade);
printf("\n add of student: %s \n", add);
getch();
```

Program 2:

Source code:

```
#include <stdio.h>
#include <conio.h>
void main()
{
```

```
float rad,area,Pie=3.14;
clrscr();
printf("Radius of circle: \n");
scanf("%f", &rad)
area=Pie * radius * radius;
printf("Area of the circle: %.f \n", area);
getch();
```

}

Output :

Radius of the Circle :

0.1

Area of the Circle :

0.0314

~~Final
11/10/17~~

OUTPUT:

Enter 1st number: 8

Enter 2nd number: 2

Addition of 2 numbers: 10

Subtraction of 2 numbers: 6

Multiplication of 2 numbers: 16

Division of 2 numbers: 4

PRACTICE NO. 2

Aim: Write a C program which will show the use of various different type of operator

#Arithmetic Operators

Source code

```
#include <stdio.h>
#include <conio.h>
void main()
{
    int num1, num2, add, sub, mul, div;
    clrscr();
    printf("Enter 1st number: ");
    scanf("%d", &num1);
    printf("Enter 2nd number: ");
    scanf("%d", &num2);
    add = num1 + num2;
    printf("Addition of 2 numbers: %d \n", add);
    sub = num1 - num2;
    printf("Subtraction of 2 numbers: %d \n", sub);
    mul = num1 * num2;
    printf("Multiplication of 2 numbers: %d \n", mul);
    div = num1 / num2;
    printf("Division of 2 numbers: %d \n", div);
    getch();
}
```

Logical operators

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

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

```
void main()
```

```
{
```

```
int x, y, z, value1, value2, value3, value4, value5;  
clrscr();
```

```
printf("Enter 1st value:");
```

```
scanf("%d", &x);
```

```
printf("Enter 2nd value:");
```

```
scanf("%d", &y);
```

~~printf("Enter 3rd value:");~~~~scanf("%d", &z);~~~~Value1 = (x < y) & & (z > y);~~

```
printf("Value1 is : %d \n", value1);
```

~~value2 = (x == y) & & (z < y);~~

```
printf("Value2 is : %d \n", value2);
```

~~value3 = (x < y) || (z == y);~~

```
printf("Value3 is : %d \n", value3);
```

~~value4 = !(x == y);~~

```
printf("Value4 is : %d \n", value4);
```

~~value5 = (x == y);~~

```
printf("Value5 is : %d \n", value5);
```

```
getch();
```

3

OUTPUT :

Enter 1st value : 9

Enter 2nd value : 8

Enter 3rd value : 2

Value 1 is : 0

Value 2 is : 1

Value 3 is : 1

Value 4 is : 0

Value 5 is : 1

OUTPUTS

The biggest number is 100

```
# Ternary operator  
#include < stdio.h>  
#include <iostream>  
void main()
```

```
{  
    int a=100, b=20, c=50, big;  
    class{  
        big = a>b ? a>c ? a : b;  
        printf("The biggest number is %d", big);  
        getch();  
    }  
}
```

PRACTICAL - 3

Aim: Decision Making Statements.

Write a program to find out odd & even numbers.

ALGORITHMS:

Step 1: Start

Step 2: [Take input] Read a number from the user

Step 3: Check if number $\% 2 == 0$

 then print even number

Step 4: EXIT

Source code :

```
#include <stdio.h>
#include <conio.h>
Void main()
{
    int n;
    clrscr();
    printf("Enter a number : ");
    scanf("%d", &n);
    if (n % 2 == 0)
    {
        printf("Even number!");
    }
    else
    {
        printf("Odd number!");
    }
    getch();
}
```

OUTPUT :-

32

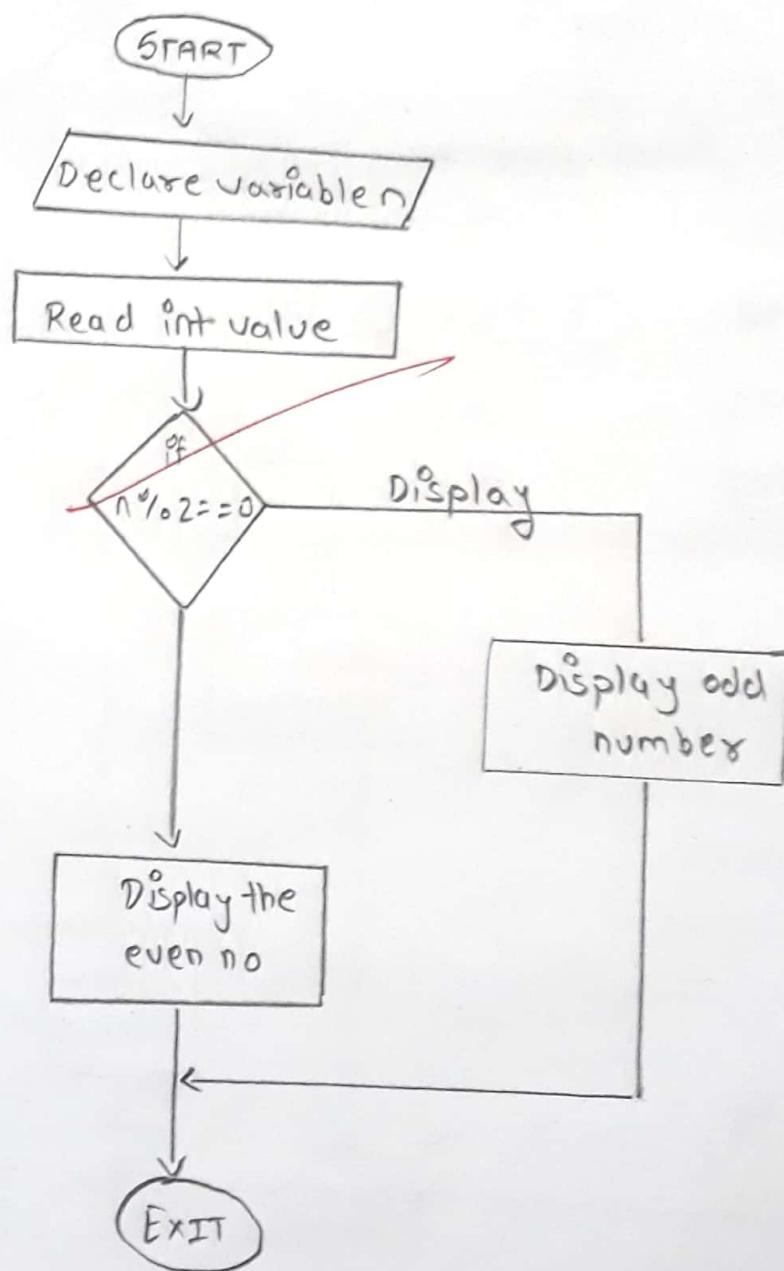
Enter a number : 26

Even number

Enter a number : 53

Odd number

Flow chart :-



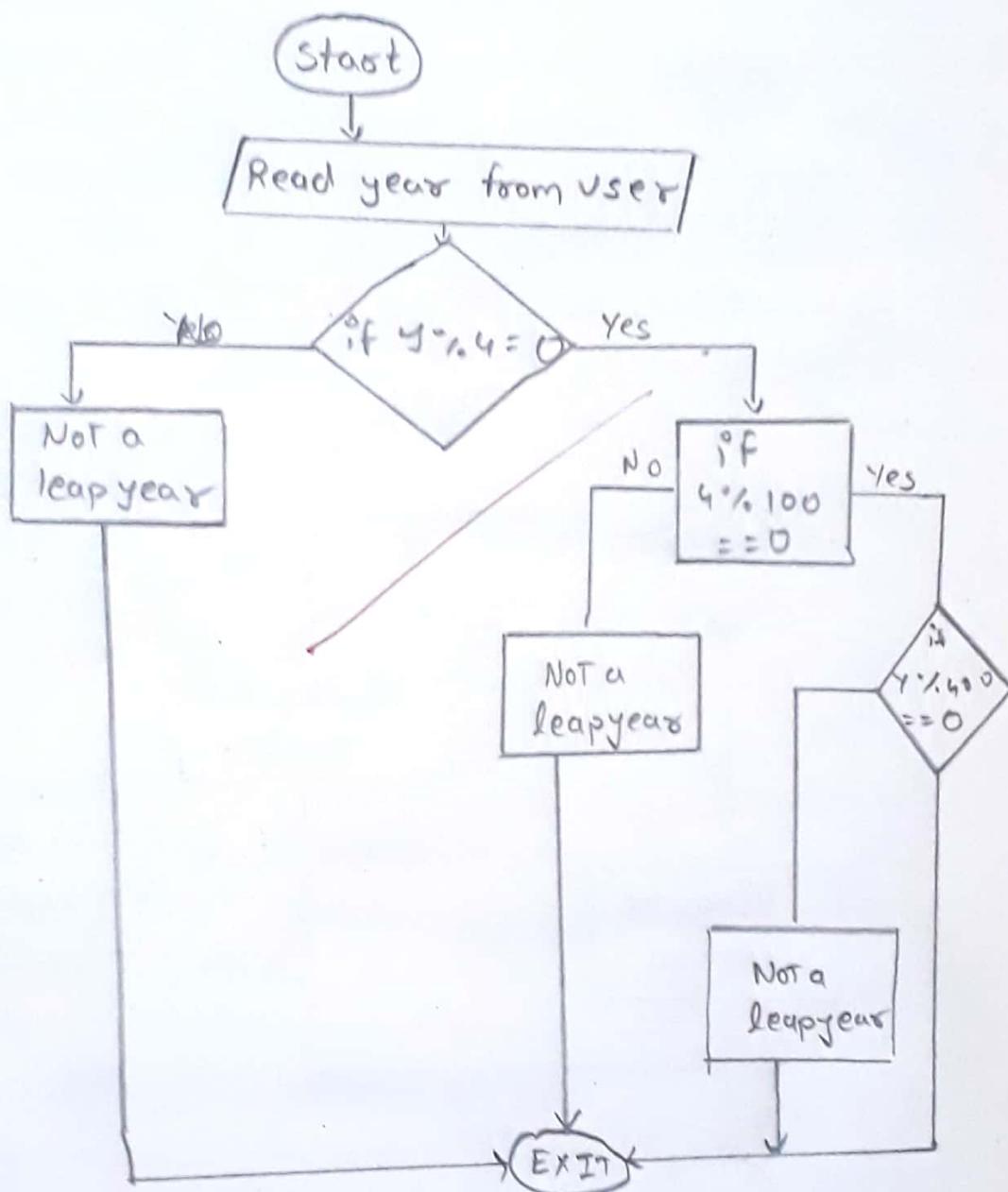
eg:-

OUTPUT:

Enter a year: 2014
Not a leap year

Enter a year: 2020
leap year

Flowchart:



Write a program to find the entered year is a leap year or not:

ALGORITHM:

Step 1: Start

Step 2: Take input] Read year from the user

Step 3: If $\text{year} \% 4 = 0$ and $\text{year} \% 400 = 0$ or
 $\text{year} \% 4 = 0$ and $\text{year} \% 100 \neq 0$
 Print not a leap year

Step 4: EXIT

Source code:

#include <stdio.h>

#include <conio.h>

{

int year;

clrscr();

printf("Enter a year:");

scanf("%d", &year);

if (year % 4 == 0)

printf("Enter a year:");

scanf("%d", &year);

if (year % 4 == 0)

if (year % 100 == 0)

{

if (year % 400 == 0)

{

```
    printf("leap year")
  }
else
  {
    printf("NOT A leap year")
  }
}

else
  {
    printf("NOT a leap year")
  }
}

else
  {
    printf("NOT a leap year")
  }

getch();
```

Q8.

Output :-

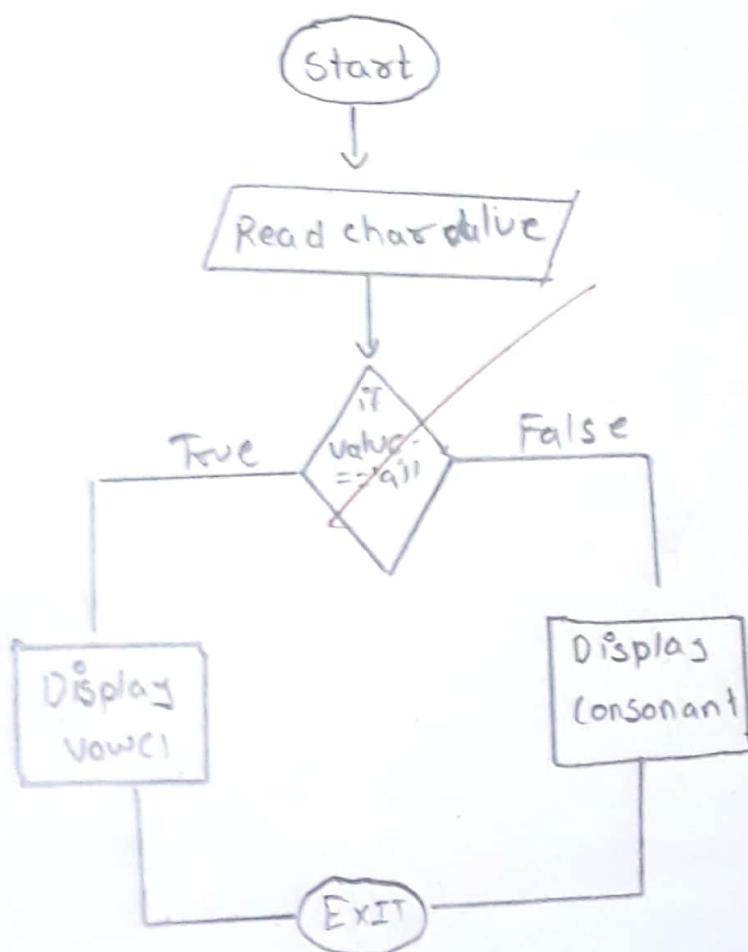
Enter a alphabet : O

Value !

Enter a alphabet : X

consanant

Flow chart :-



write a program to find whether the character is vowel or consonant

ALGORITHM :

Step 1 : Start

Step 2 : [Take Input] Read characters value from user.

Step 3 : [Check] If value == 'a' || value == 'e' ||
 value == 'i' || value == 'o' || value == 'u' ||
 value == 'A' || value == 'E' || value == 'I' ||
 value == 'O' || value == 'U'

Step 4 : Exit

Source Code :

```
#include <stdio.h>
#include <conio.h>
void main()
{
    char a;
    clrscr();
    printf("Enter the alphabet : ");
    scanf("%c", &a);
    if (a == 'a' || a == 'e' || a == 'i' || a == 'o' ||  

        a == 'u' || a == 'A' || a == 'E' || a == 'I' ||  

        a == 'O' || a == 'U')
    {
        printf("Vowel");
    }
    else
    {
        printf("Consonant");
    }
    getch();
}
```

PRACTICAL No. 4

Aim: Write a program to print even numbers between 1-50 using while loop.

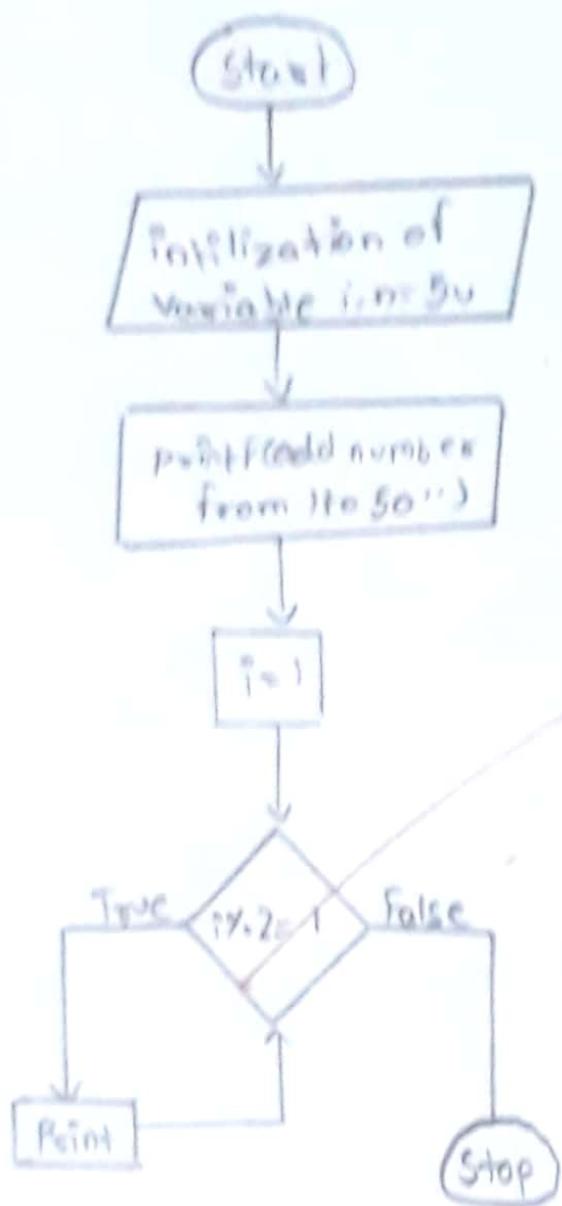
a) Source code:

```
#include <stdio.h>
#include <conio.h>
void main()
{
    int i, n = 50;
    clrscr();
    printf("All even numbers from 1 to 50 are : \n");
    i = 2;
    while (i <= n)
    {
        printf("%d \n", i);
        i = i + 2;
    }
    getch();
}
```

Output:
All even number from 1 to 50 are

2
4
6
8
10
12
14
16
18
20
22
24
26
28
30
32
34
36
38
40
42
44
46
48
50

pg



Algorithm :

Step

- 1) Start
- 2) Initialize two variable with static variable where $n = 50$ & $i = 2$
- 3) Use while loop for printing the even number upto the range 50.
- 4) Adding 2 to current even number will give next even number.
- 5) Display the appropriate output
- 6) Stop.

b) Aim: Write a C program to print odd number between 1-50 using while do loop.

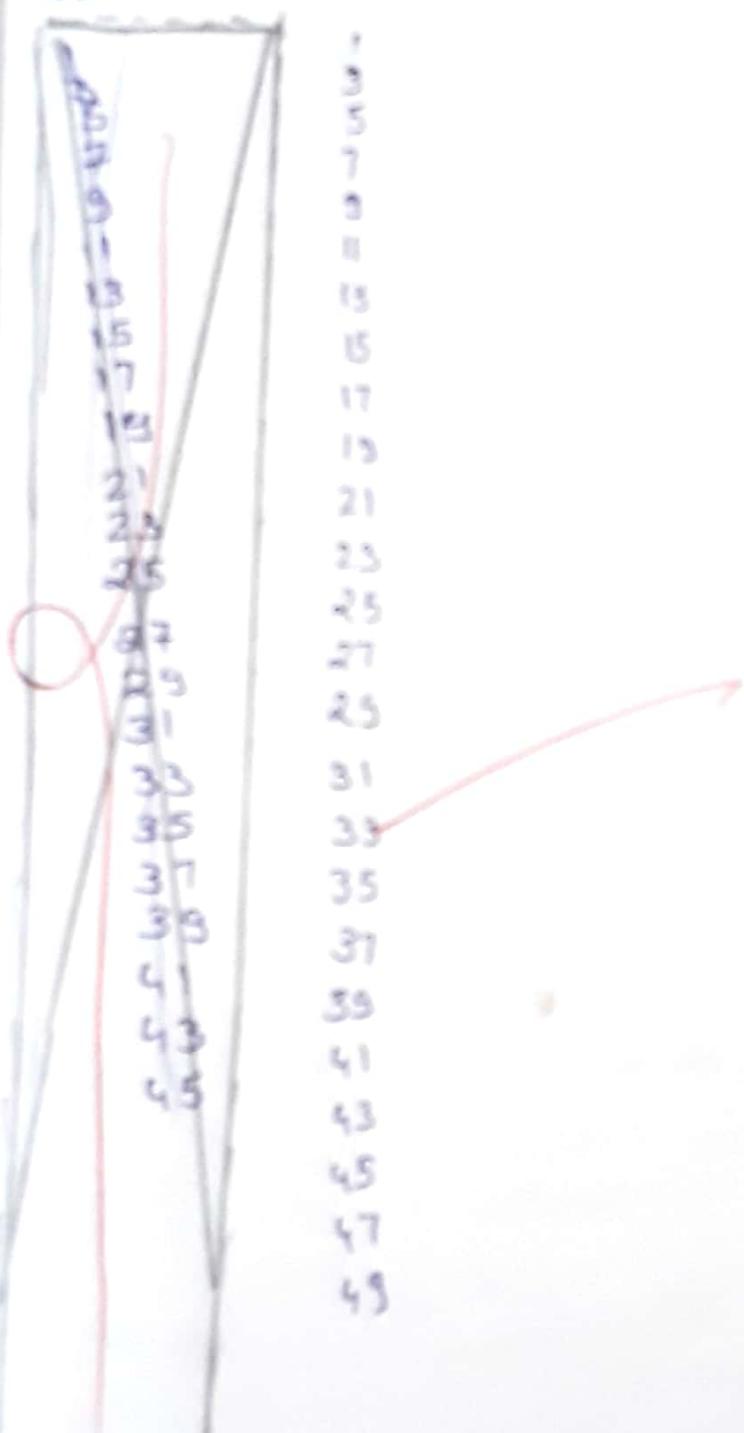
Source code:

```
#include <stdio.h>
#include <conio.h>
Void main()
{
    int i, n = 50;
    clrscr();
    printf (" odd numbers from 1 to 50 are: \n", n);
    i = 1;
    do
    {
        if (i % 2 == 1)
        {
            printf ("%d \n", i);
        }
        i++;
    }
    while (i <= n);
    getch();
}
```

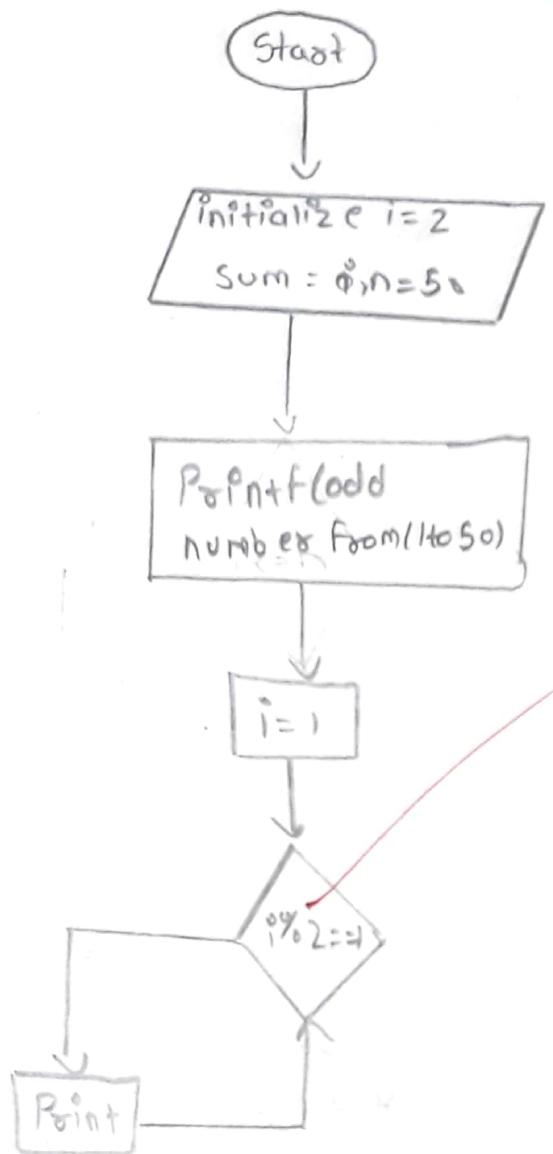
OUTPUT :

odd number from 1 to 50 are

38



28



Algorithm:

Step.

- 1) Start
- 2) Initialize two static variable $n=50, i=1$.
- 3) Use do while loop for iterations from 1 to 50
- 4) Use if condition statement to check whether given number is even or odd.
- 5) Increment the value of i .
- 6) Display the appropriate output
- 7) Stop

c) Aim: Write a C program to print sum of all numbers between 1 to n using for loop.

Source code:

```
#include <stdio.h>
#include <conio.h>
void main()
{
    int i, n, sum = 0;
    clrscr();
    printf("Enter the range = ");
    scanf("%d\n", &n);
    for (i = 2; i <= n; i = i + 2)
    {
        sum = sum + i;
    }
    printf("Sum of all even numbers upto the
range are = ", sum);
    getch();
}
```

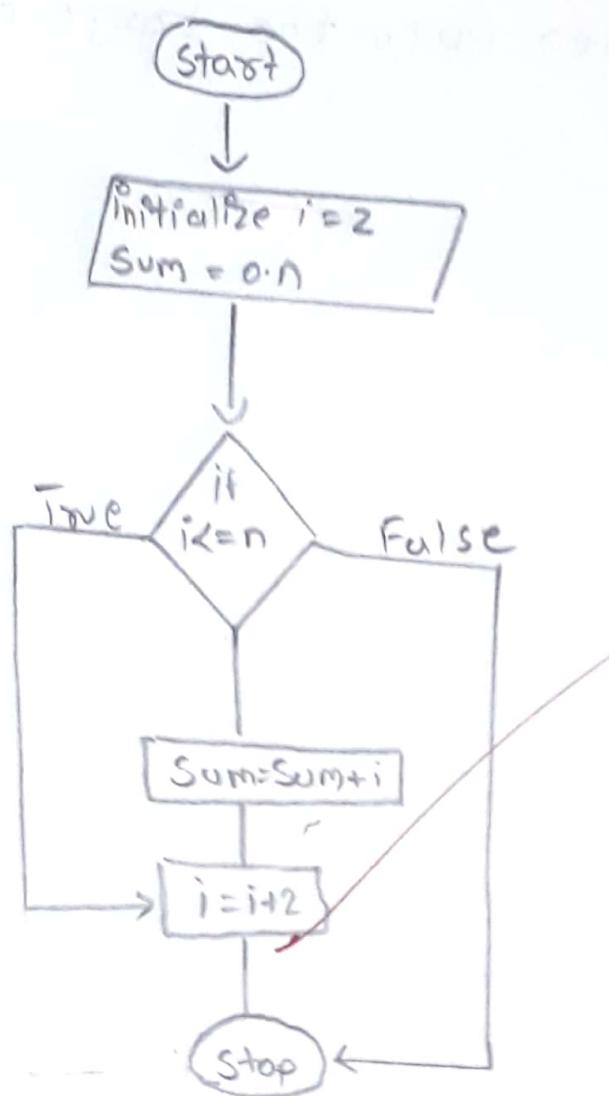
Output:

Enter the range = 10

Sum of all even numbers upto the range are 30

40

Q8



Algorithm :-

STEPS

- 1) Start
- 2) Initialize three variable of ~~the~~ and one is static and two
3) is dynamic
- 3) Use for loops for check the the given range.
- 4) Add current even number.
- 5) Display the appropriate.
- 6) Stop

*Sri,
Jai Tosh 2020*

PRACTICAL No. 5

A) Aim: WAP to print input of array elements.

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

```
main()
```

```
{
```

```
    int a[20];
    int size, i;
    printf ("Enter any no : ");
    scanf ("%d", &size);
    for (i=0; i<size; i++)
```

```
{
```

```
    printf ("Enter the a[%d] no. element ", i);
    scanf ("%d", &a[i]);
}
```

```
printf ("\n Elements in array are : ");
for (i=0; i<10; i++)
```

```
{
```

```
    printf ("%d ", arr[i]);
}
```

```
clrscr();
getch();
```

```
3
```

OUTPUT :

Enter any no : 20

Enter the no. of element : 10

0

1

2

3

4

5

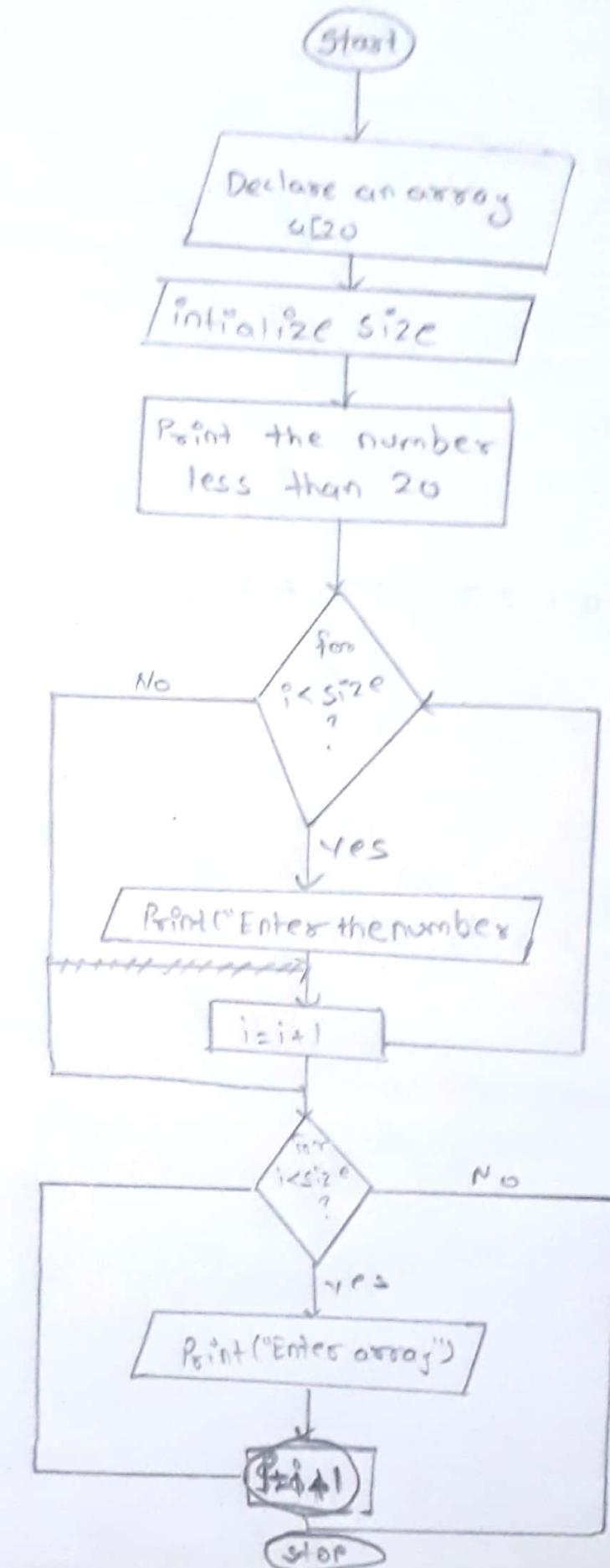
6

7

8

9

Elements in array are: 0 1 2 3 4 5 6 7 8 9



Algorithm :

- 1) Read the size of an array and store that value into the variable n.
- 2) Scanf() function reads the entered element and initialize that element to a[i] until all iterations of for loop are scanf ("%d", &a[i]) using for (i=0; i<n; i++)
- 3) Print the array element of a[] using for loops which iterates i=0 to i<size of an array.
- 4) printf ("%d", a[i]), it prints the elements of an array from i=0 to i<n using for loop.

B)

WAP to find sum of array elements

SOURCE CODE:

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

int main()
{
    int a[100];
    int i,n,sum=0;
    printf("Enter size of the array:");
    scanf("%d", &n);

    printf("Enter a[%d] elements in the array:", n);
    for(i=0; i<n; i++)
    {
        scanf("%d", &a[i]);
    }

    for(i=0; i<n; i++)
    {
        sum = sum + a[i];
    }

    printf("Sum of all element of array = %d", sum);
    return 0;
}
```

3

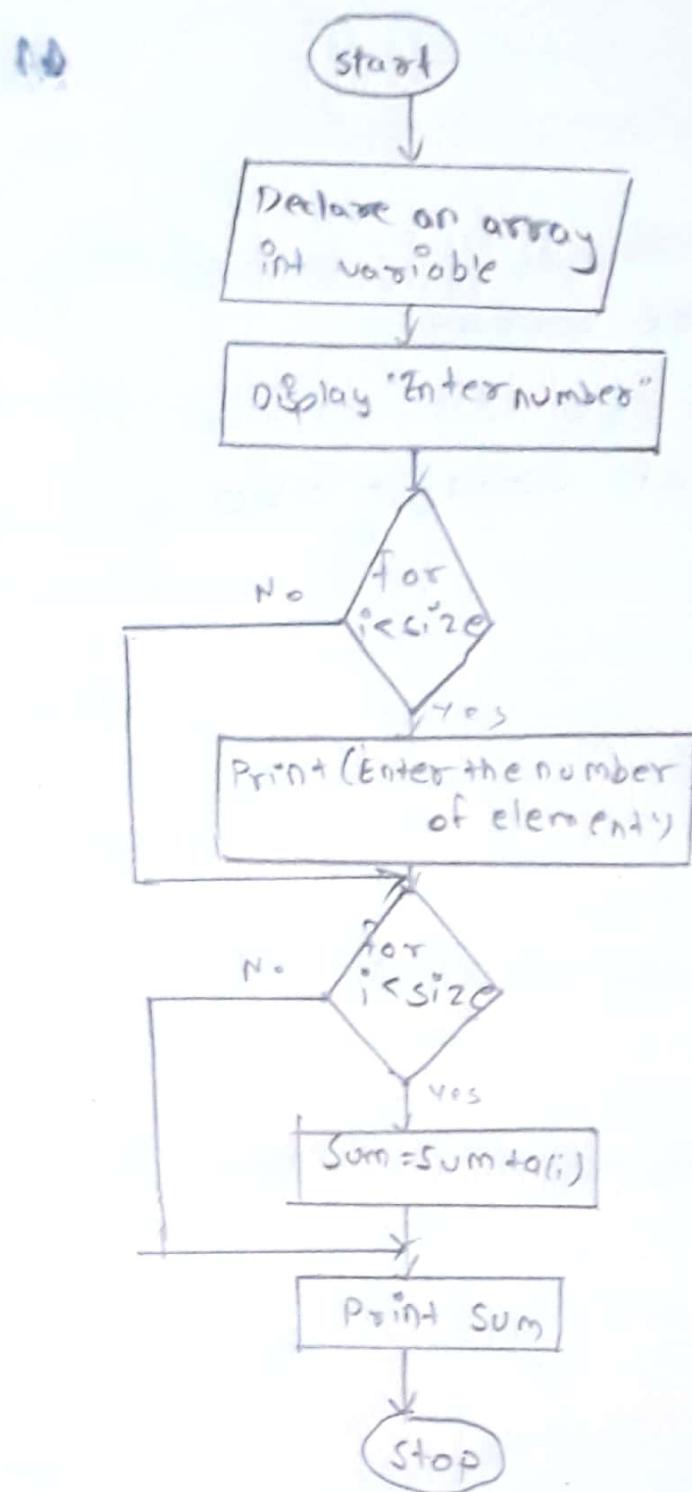
OUTPUT:

Enter size of the array: 10

Enter 10 elements in the array:

10 20 30 40 50 60 70 80 90 100

sum of all elements of array = 550



Algorithm

- 1) Input size and elements in array, store in some variable say n and a[n]
- 2) To store sum of array elements, initialize a variable sum = 0. Note: sum must be initialized only with 0.
- 3) To find sum of all elements, iterate through each element and add the current element to the sum. which is run a loop from 0 to n. the loop structure should look like
`for(i=0; i<n; i++)`.
- 4) Inside the loop add the current array element to sum i.e. $sum = sum + a[i]$ or even you can do $sum += a[i]$

Q. Write a program to find out fibonacci series using %d, %f, %c

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

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

```
void main()
```

```
{
```

```
    int a[20], n, i, j;
```

```
    printf("Enter the no. of terms:");
```

```
    scanf("%d", &n);
```

```
    a[0] = 0, a[1] = 1;
```

```
    for (i=2; i<n; i++)
```

```
{
```

```
        a[i] = a[i-1] + a[i-2];
```

```
}
```

```
    printf("The fibonacci series upto n term is %.d", n);
```

```
    for (i=0; i<n; i++)
```

```
{
```

```
        printf("%d\t", a[i]);
```

```
}
```

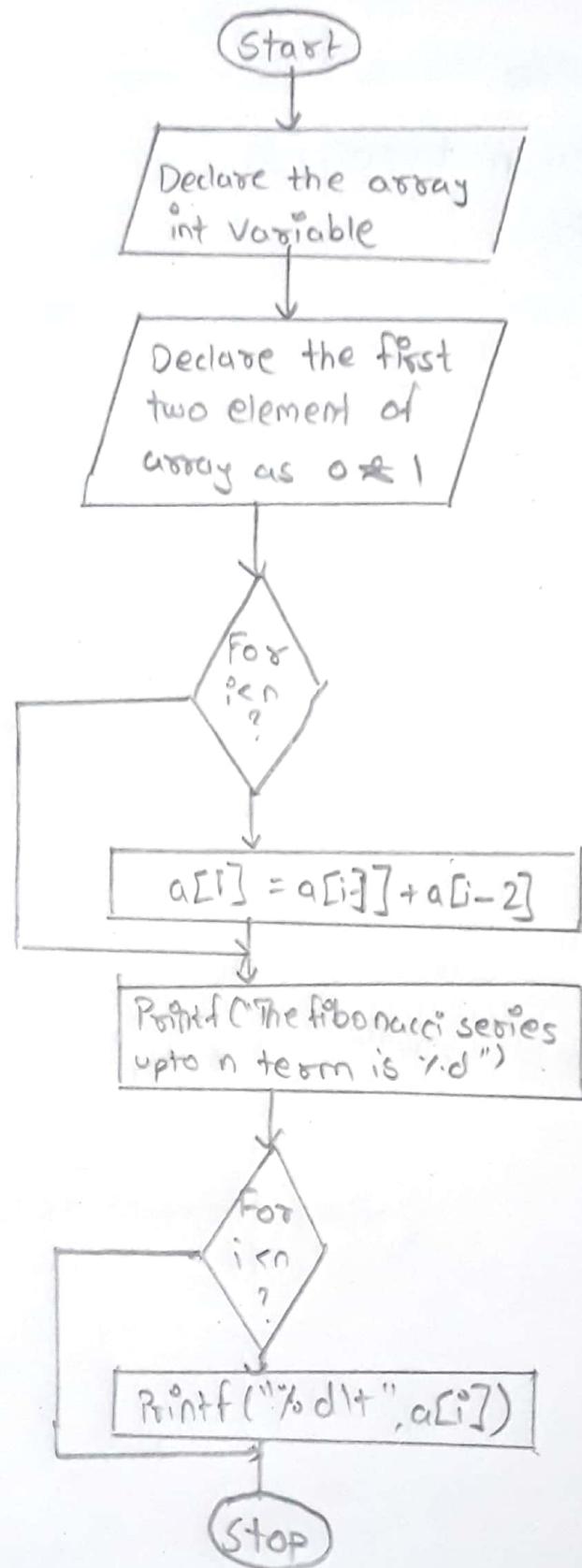
```
}
```

Output:

Enter the no. of terms: 10

The fibonacci series upto n term is

0 1 1 2 3 5 8 13 21 34



Algorithm:

- 1) Declare a array of anysize of data type int
- 2) Accept a value from user till you want to display the fibonacci series.
- 3) Initialize first element of array to 0 and second element to 1 as series starts from 0 & 1.
- 4) Use for loop to develop fibonacci series.
- 5) Display the series using print() function.

54

PRACTICAL No. 6.

- a) Aim: (Program to print individual digits of entered number.)

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

```
void main()
```

```
{
```

```
    long number = 0;
```

```
    int temp-var = 0;
```

```
    int i = 1;
```

```
    clrscr();
```

```
    printf("In Enter a number : ")
```

```
    scanf("%i", &number);
```

```
    while (number >= 1)
```

```
{
```

```
        temp-var = number % 10;
```

```
        printf("\n\n Digit at %d place => %d ", i, temp-var);
```

```
        number = number / 10
```

```
        i++;
```

```
}
```

```
getch()
```

```
3
```

Output :

48

Enter a number : 175481

Digit at 1 place \Rightarrow 1

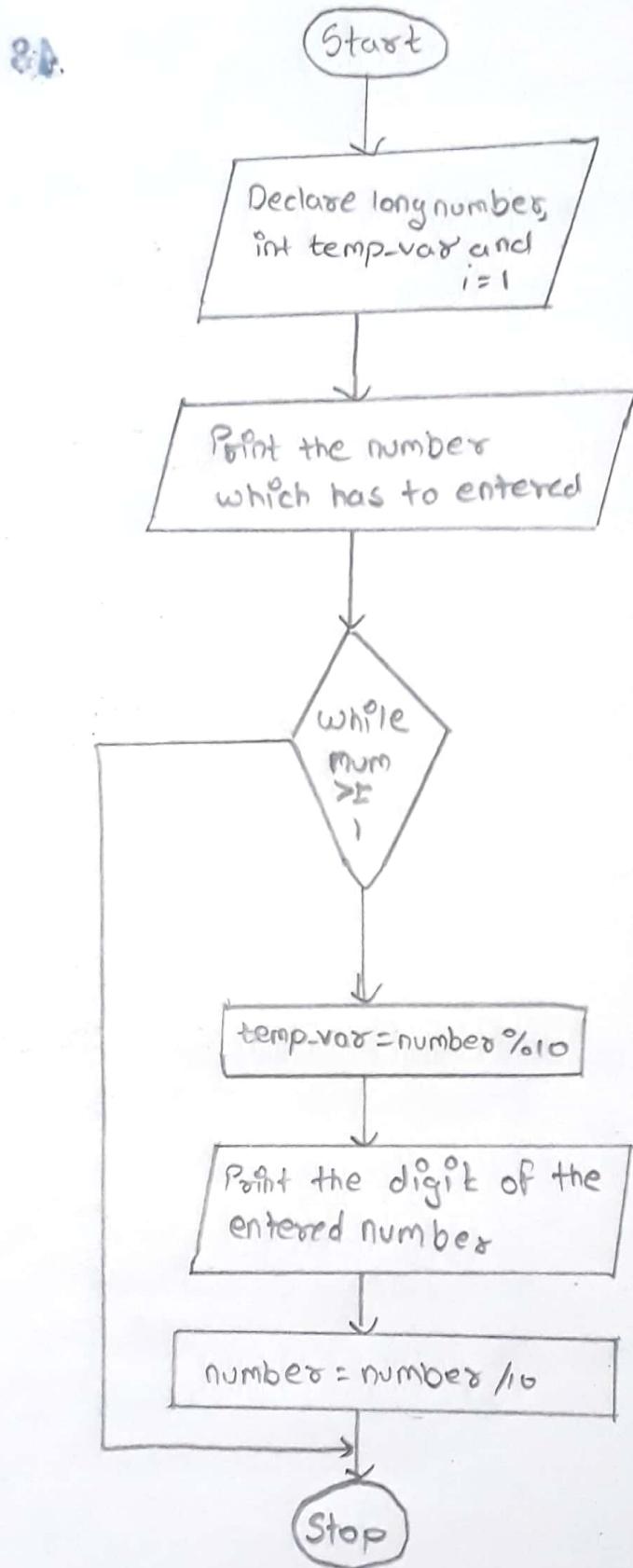
Digit at 2 place \Rightarrow 8

Digit at 3 place \Rightarrow 4

Digit at 4 place \Rightarrow 5

Digit at 5 place \Rightarrow 7

Digit at 6 place \Rightarrow 1



Algorithm

Step 1: Start

Step 2: Declare the variable

Step 3: Use the printf() and scanf() for printing the number

Step 4: Use the while loop for getting the digit at unit place i.e. temp-var = number % 10 then print for removing the unit place digit and for moving backward use number = number / 10

Step 5: Stop.

(a)

Aim: Average of 3 entered numbers.

```
#include<stdio.h>
#include<conio.h>
void average(int sum);
void sum(int x, int y, int z);
void main()
{
    clrscr()
    int a, b, c
    printf ("\n Enter the value of a,b,c => ");
    scanf ("%d %d %d", &a, &b, &c);
    sum(a, b, c);
    getch();
}

void sum(int x, int y, int z)
{
    int s;
    s = a+b+c;
    average(s);
}

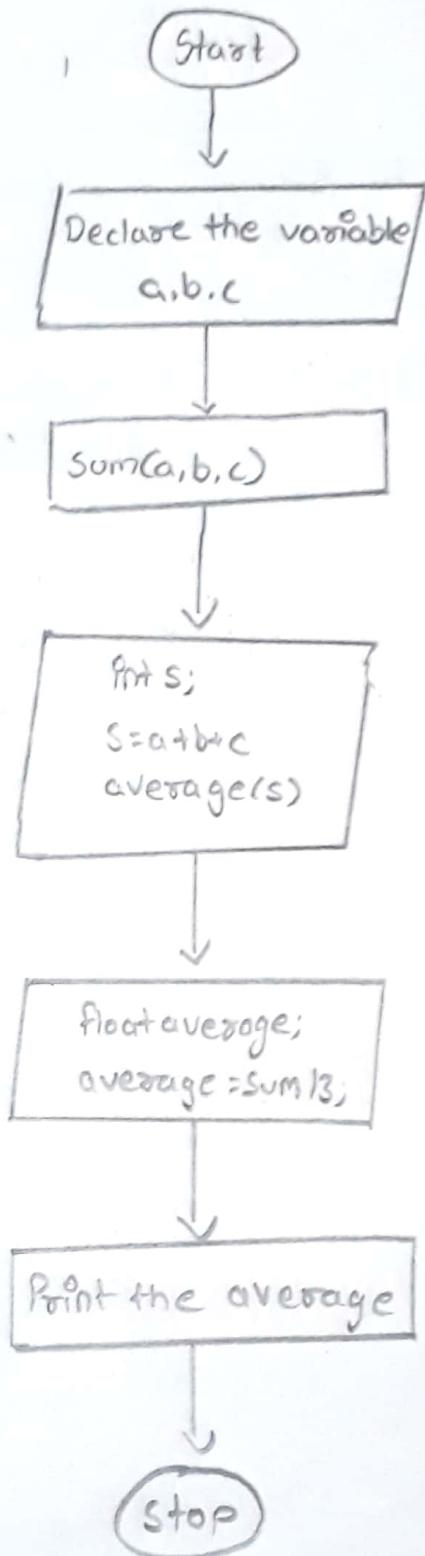
void average(int sum)
{
    float average;
    average = sum/3;
    printf ("\n Average => %f", avg);
}
```

Output :

Enter value a,b,c => 5 6 8

Average => 6.3333

Q2



Algorithm.

Step 1: Start

Step 2: Declare the variables.

Step 3: Use the Sum function and find the sum
of three entered values.

Step 4: Now declare the average function and divide the
the sum by 3 and print the average of
3 entered numbers.

Step 5: Stop.

Q) WAP to find factorial of a number using recursion function.

```
#include < stdio.h>
#include < conio.h>
int factorial(int n);
void main()
{
    clrscr();
    int x, fact;
    printf ("In Enter value of x:");
    scanf ("%d", &x);
    fact = factorial(x);
    printf ("In Factorial of %d = %d", x, fact);
    getch();
}

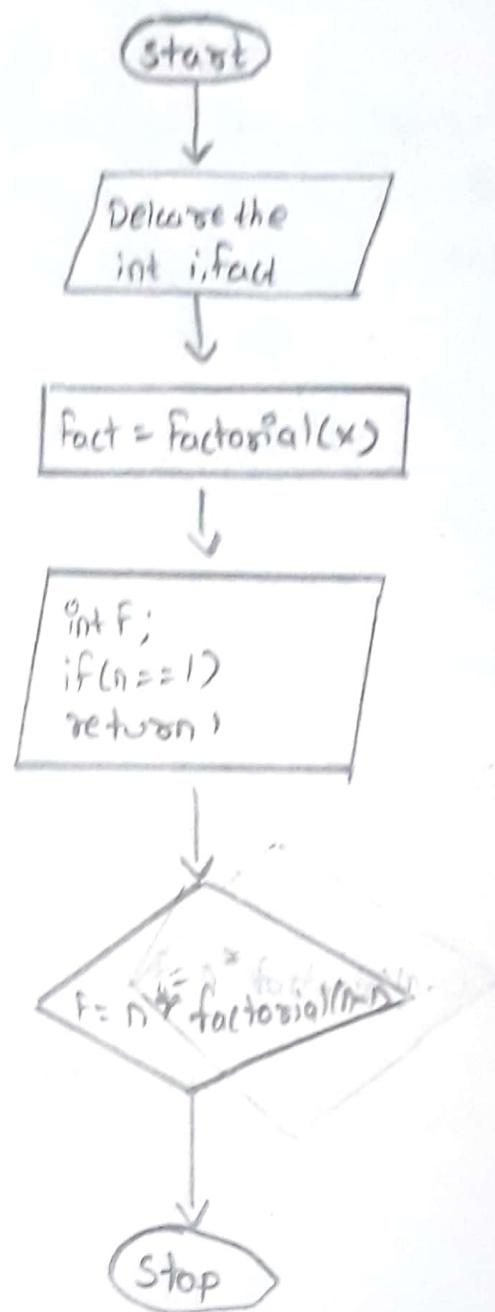
int factorial(int n)
{
    int f;
    if (n == 1)
        return (1)
    else
        f = n * factorial (n-1);
    return (f)
}
```

Output :

Enter the value of x : 8

factorial of 8 = 40320

28



Algorithm :

Step 1 : start

Step 2 : Read number n

Step 3 : call factorial(n)

Step 4 : print factorial f

Step 5 : call factorial(n)

Step 6 : if $n=1$ then return 1

Step 7 : else

$$f = n * \text{factorial}(n-1)$$

Step 8 : Return f

Step 9 : Stop.