

Output:

SD Demonstrate Various datatypes

Name of student:

Gaurav

Address of the student

Mumbai

Roll no of student

32

Percentage of the student

75

Grade of student

B

Mobile No

0987654321

Student name: Gaurav

Student address: Mumbai

Student roll no: 32

Student percent: 75

Student grade: B

Student mobile no: 0987654321

## Practical - 1

Aim :- To study the use of different types of datatypes :

source code :

```
# include < stdio.h >
# include < conio.h >
void main()
{
    char name[50];
    char add[50];
    int rollno;
    float percent;
    char grade;
    long int mob;
    clrscr();
    printf("Demonstrate various datatypes");
    printf("Name of the student\n");
    scanf("%s", &name);
    printf("Address of the student\n");
    scanf("%s", &add);
    printf("Roll no of the student\n");
    scanf("%d", &rollno);
    printf("Percentage of student\n");
    scanf("%f", &percent);
}
```

ESD

```
printf("Grade of student\n");
scanf("%s", &grade);
printf("Mobile no\n");
scanf ("%d", &mob);
printf ("\n Student name: %s", name);
printf ("\n Student address: %s", add);
printf ("\n Student rollno: %d", rollno);
printf ("\n Student percent: %.f", percent);
printf ("\n Student grade: %.1c", grade);
printf ("\n Student mobilenr: %.d", mob);
getch();
```

3

Sriji  
29/11/19

SSO  
# output

Enter 1<sup>st</sup> no: 8

Enter 2<sup>nd</sup> no: 8

Addition is 16

Subtraction is 0

Multiplication is 64

Division is 1

Aim : Write a C Program to demonstrate the use of various types of operators

## # Arithmetic operators

# include &lt; stdio.h &gt;

# include &lt; conio.h &gt;

void main()

{

int num1, num2, add, sub, mul, div;  
clrscr();printf("Enter 1<sup>st</sup> No: ");

scanf("%d", &amp;num1);

printf("Enter 2<sup>nd</sup> No: ");

scanf("%d", &amp;num2);

add = num1 + num2;

printf("Addition is: %.d", add);

sub = num1 - num2;

printf("In Subtraction is %.d", sub);

mul = num1 \* num2;

printf("In multiplication is %.d", mul);

div = num1 / num2;

printf("In Division is %.d", div);

getch();

}

ISO

## # 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 :

028

Enter 1<sup>st</sup> Value : 9

Enter 2<sup>nd</sup> Value : 5

Enter 3<sup>rd</sup> Value : 2

Value 1 is : 0

Value 2 is : 1

Value 3 is : 1

Value 4 is : 0

Value 5 is : 1

*Jmij  
24/10/2020*

## Output:

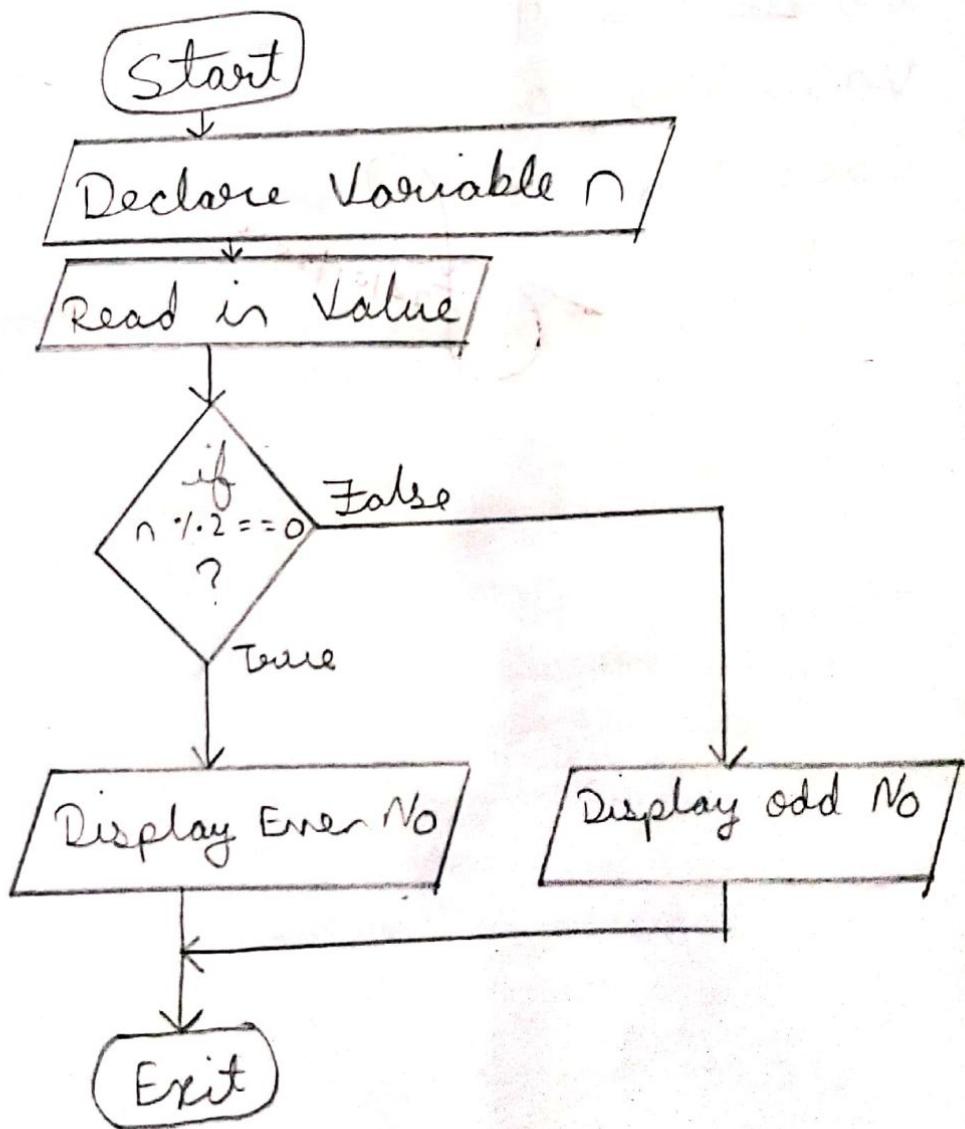
Enter a number : 46

Even number

Enter a number : 67

Odd Number

## Flowchart:



## Practical - 3

Aim : Programs on Decision Statement.

i) Write a program to find odd & even numbers.

Algorithm :

- Step 1 : Start
- Step 2 : [Take Input] Read a number from user.
- Step 3 : Check if  $n \% 2 == 0$  Then print "Even Number" else print "Odd Number".
- Step 4 : Exit

Program :

```
# 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");
    }
}
```

CSO

getch();  
3

- 2) Write a program to find the entered year  
is leap year or Not.

Algorithm :

Step 1 : Start  
Step 2 : [Take Input] Read year from user.  
Step 3 : If  $\text{year} \% 4 = 0$  and  $\text{year} \% 400 == 0$  OR  
 $\text{year} \% 4 = 0$  and  $\text{year} \% 100 != 0$   
print "leap year".  
else print "Not a leap year".  
Step 4 : Exit.

Program:

```
#include <stdio.h>
#include <conio.h>
void main()
{
    int year;
    clrscr();
    printf("Enter a Year:");
    scanf("%d", &year);
    if (year \% 4 == 0)
    {
        if (year \% 100 == 0)
            if (year \% 400 == 0)
```

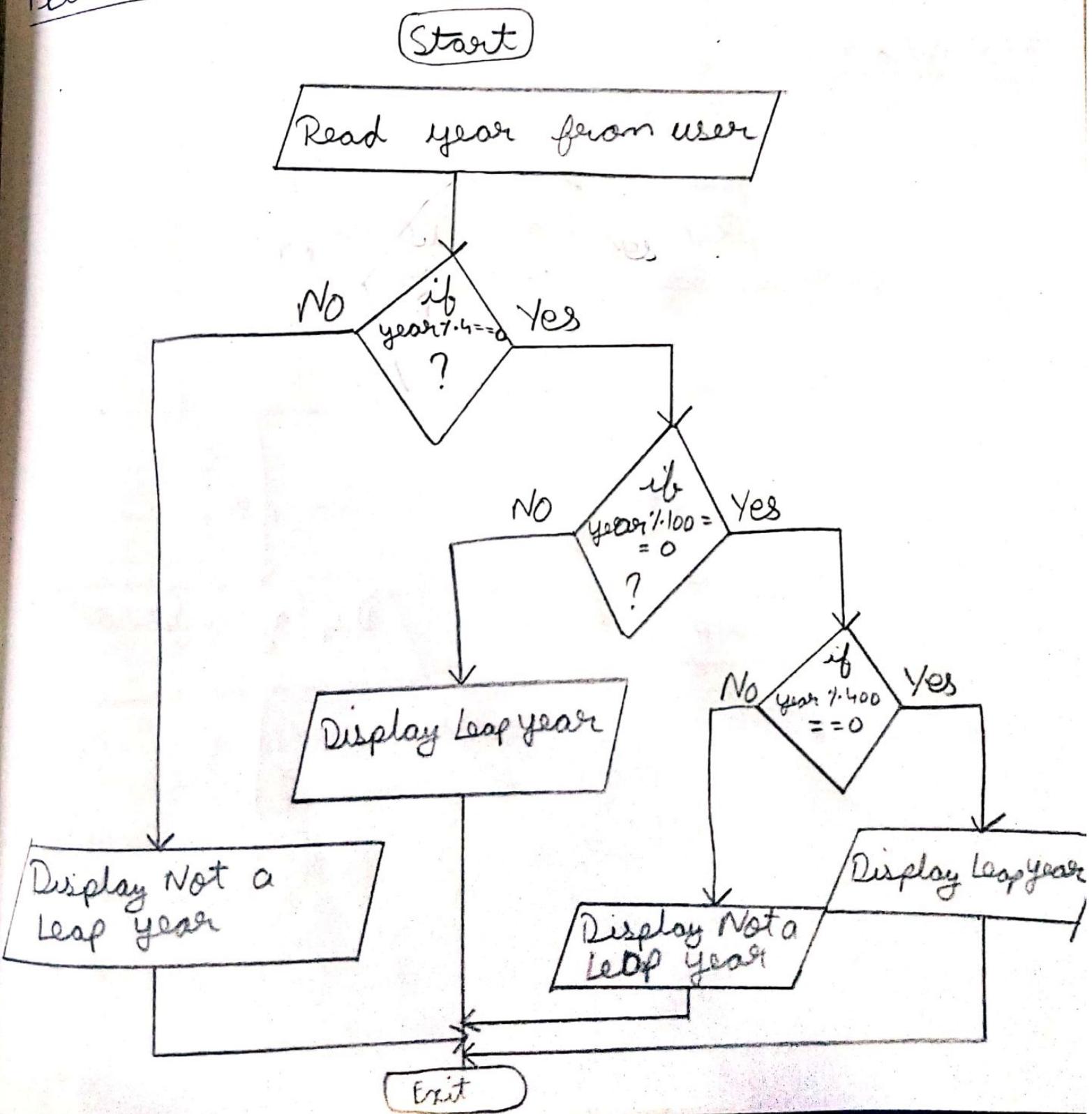
## Output :

Enter a year : 2017  
Not a leap year

030

Enter a year : 2020  
leap year

## Flowchart :



Output

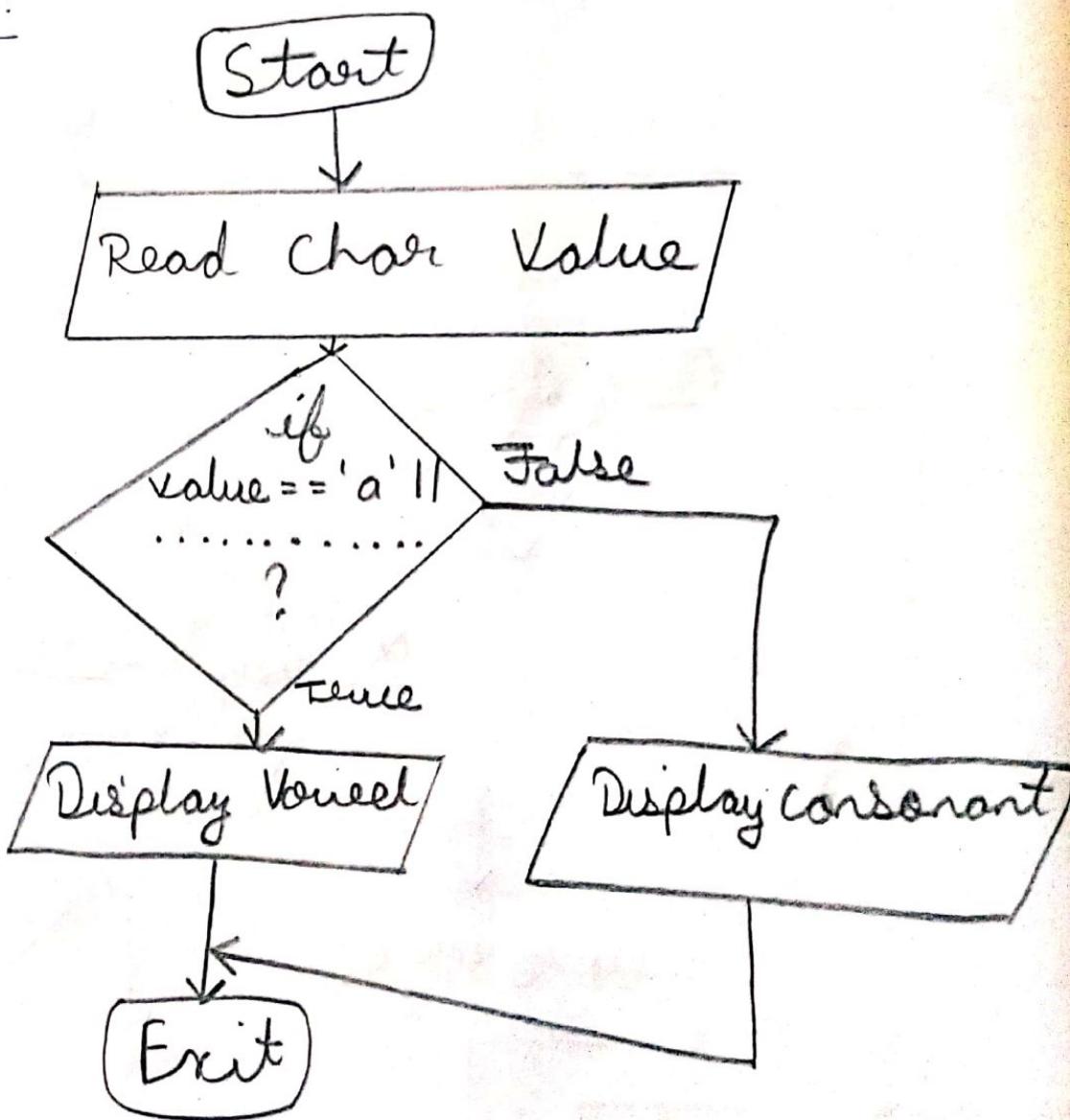
180  
Enter the Alphabet: o

Vowel.

Enter the Alphabet: g.

Consonant.

Flowchart:



```

    {
        printf("Leap Year");
    }
    else
    {
        printf("Not a leap year");
    }
    else
    {
        printf("Leap year");
    }
    else
    {
        printf("Not a leap year");
    }
getch();
}

```

3) Write a program to find whether the entered character is vowel or consonant.

Algorithm:

Step 1 : start

Step 2 : [Take Input] Read character 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 == 'O'

print "Vowel"  
 else print "Consonant"  
 Step 4 : Exit

Program :

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

Output:

All even numbers 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

## Practical 4

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

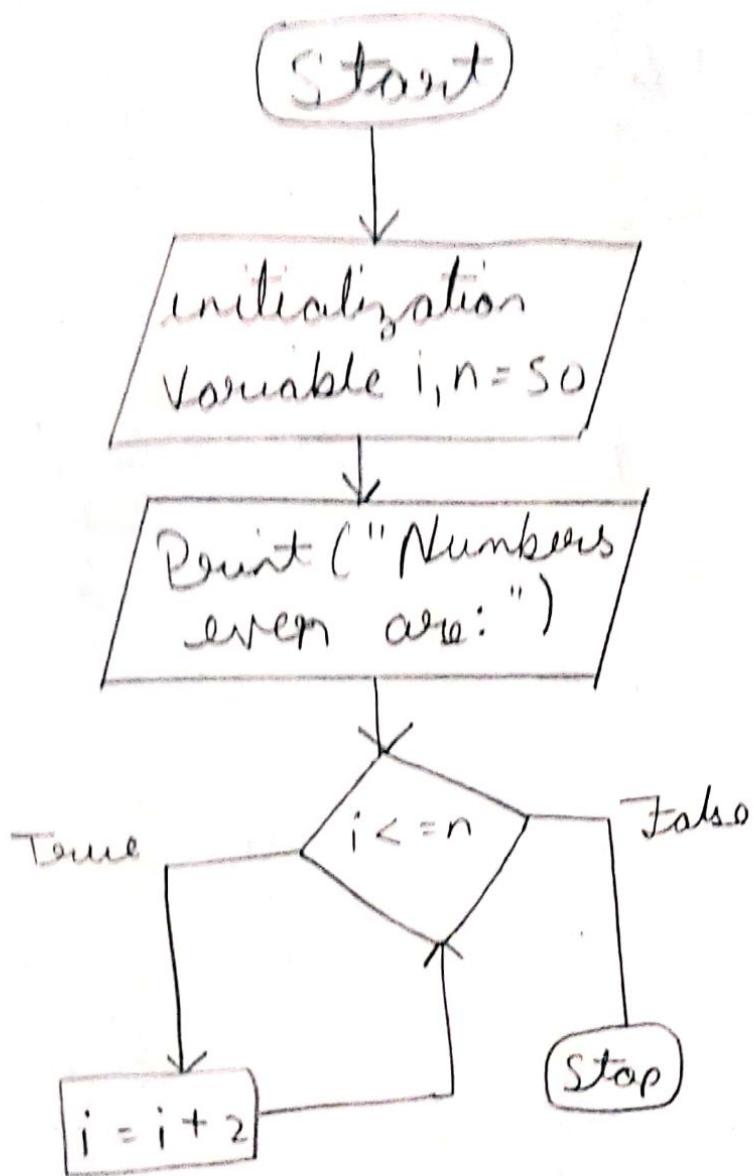
Source code:

```
#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);
        getch();
    }
}
```

## 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



Output

odd numbers from 1 to 50 are:

1  
3  
5  
7  
9  
11  
13  
15  
17  
19  
21  
23  
25  
27  
29  
31  
33  
35  
37  
39  
41  
43  
45  
47  
49

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

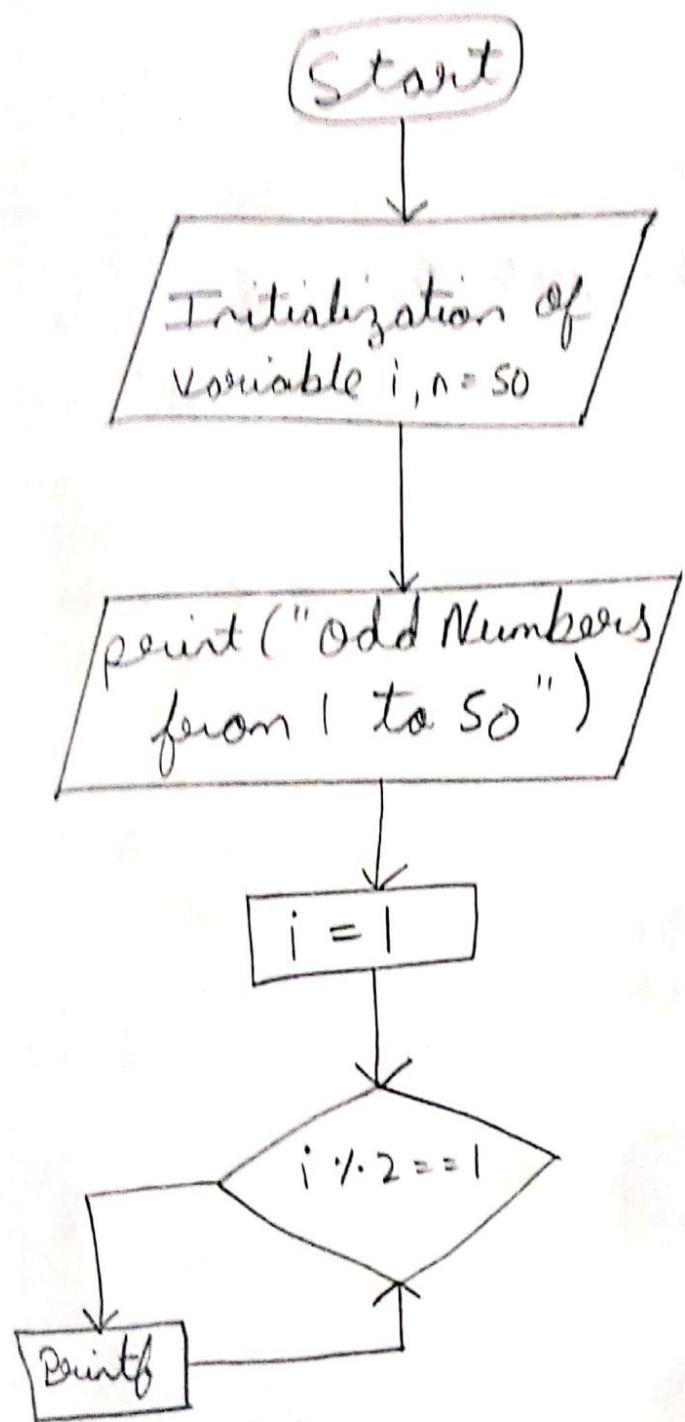
Source code:

```
#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();
}
```

Algorithm

Step :

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



~~780~~  
Output :

Enter the range 10

Sum of the even numbers upto the range are : 30

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

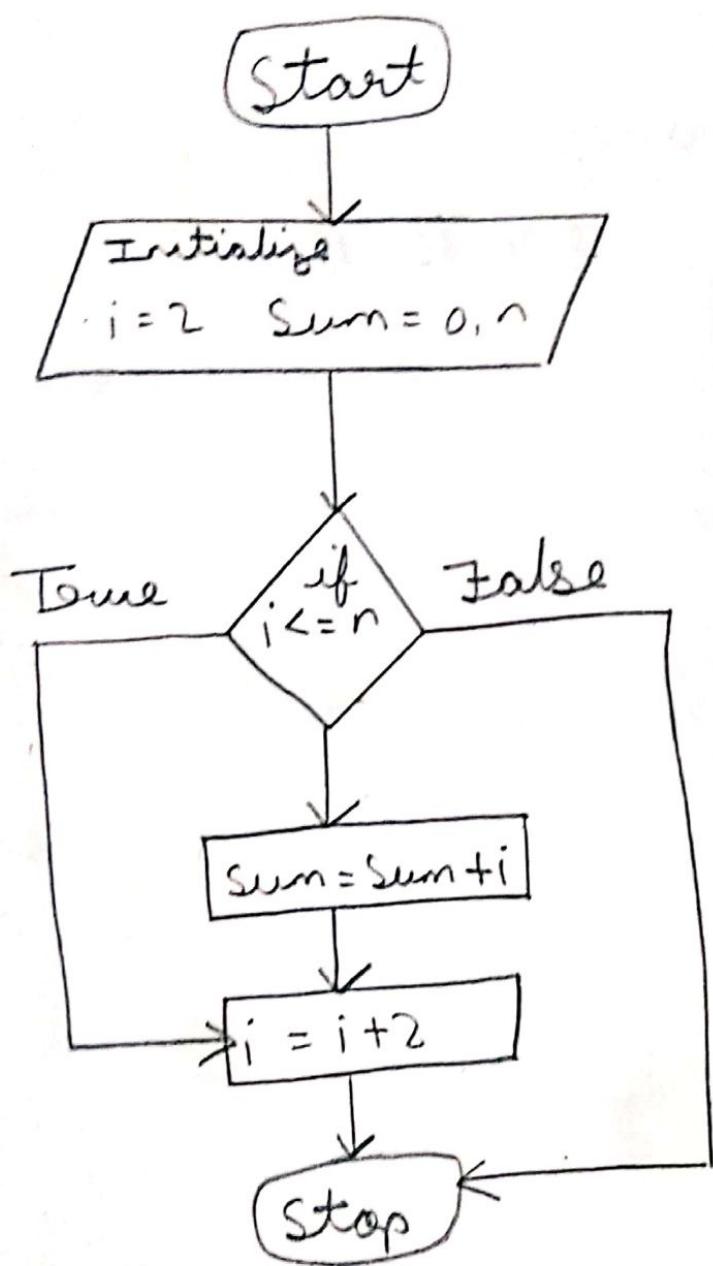
Source code:

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

## Algorithm :

Steps :

- 1) Start
- 2) Initialize three variable from these two  
static and one is dynamic.  
 $i = 2$ ;  $sum = 0$ ;  $n$ ;
- 3) Use for loop for check the even number &  
print upto the given range.
- 4) Add current even number to sum.
- 5) Display the appropriate output.
- 6) Stop



Output :-

Enter 10 data of array:

1 2 3 4 5 6 7 8 9 10 11

largest = 11

## Practical 5

Aim: To understand the concept of Arrays.

### One Dimensional Arrays:

Program: Find the largest number in an array of 10 numbers.

```
#include < stdio.h >
#include < conio.h >
void main()
{
    clrscr();
    int i, l, a[10];
    printf("In Enter 10 data of array:");
    for (i=0; i<10; i++)
    {
        scanf("%d", &a[i]);
    }
    l = a[0];
    for (i=1; i<10; i++)
    {
        if (l < a[i])
        {
            l = a[i];
        }
    }
    printf("In largest : %d", l);
}
```

Program: Find no. of even numbers.

```
#include <stdio.h>
#include <conio.h>
void main()
{
    int l, a[10];
    clrscr();
    printf("\nEnter 10 data of array");
    for(i=0; i<10; i++)
    {
        printf("\n%d", &a[i]);
    }
    c = 0;
    for(i=0; i<10; i++)
    {
        a = a[i] % 2;
        if(a == 0)
            c++;
    }
    printf("\n No. of even no = %d", c);
    getch();
}
```

Output :

Enter 10 data of array:

1 2 3 4 5 6 7 8 9 10

No. of even no.: 5

## ⇒ Two dimensional Array

Program 1: To add two matrix each of size  $3 \times 3$ :

```
# include < stdio.h >
# include < conio.h >
void main ()
{
    clrscr();
    int x[3][3], y[3][3], z[3][3];
    int r, c;
    printf ("Enter elements of matrix x:");
    for (r=0; r<3; r++)
    {
        for (c=0; c<3; c++)
        {
            scanf ("%d", &x[r][c]);
        }
    }
    printf ("Enter elements of matrix y:");
    for (r=0; r<3; r++)
    {
        for (c=0; c<3; c++)
        {
            scanf ("%d", &y[r][c]);
        }
    }
    for (r=0; r<3; r++)
    {
        for (c=0; c<3; c++)
        {
            z[r][c] = x[r][c] + y[r][c];
        }
    }
    printf ("Sum of two matrices is:\n");
    for (r=0; r<3; r++)
    {
        for (c=0; c<3; c++)
        {
            printf ("%d ", z[r][c]);
        }
    }
}
```

```
for (row = 0; row < 3; row++)  
    for (col = 0; col < 3; col++)  
        z[row][col] = x[row][col] + y[row][col];  
  
{  
    cout << "In Matrix z: ";  
    for (row = 0; row < 3; row++)  
        {  
            for (col = 0; col < 3; col++)  
                cout << "\t" << z[row][col];  
            cout << endl;  
        }  
    cout << endl;  
}  
catch ()
```

Output

040

Enter elements of matrix X: 0 1 2 3 4 5 6 7 8

Enter elements of matrix Y: 0 1 2 3 4, 5 6 7 8

Matrix X :    0        2        4  
                6        8        10  
                12      14      16

100  
Output :

Enter the table of marks

1

2

3

4

Total no. of columns is 4

Total no. of rows is 6.

Program 5 - Write a program to print column with addition.

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

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

```
void main ()
```

```
{
```

```
clrscr();
```

```
int r, c, total, m[5][4];
```

```
printf ("In Enter table of marks: ");
```

```
for (r=0; r<5; r++)
```

```
{
```

```
for (c=0; c<4; c++)
```

```
{
```

```
scanf ("%d", &m[r][c]);
```

```
}
```

```
}
```

```
for (c=0; c<4; c++)
```

```
{
```

```
total = 0;
```

```
for (r=0; r<5; r++)
```

```
{
```

```
total = total + m[r][c];
```

```
}
```

```
}
```

```
printf ("In Total of Column is: %d", c, total);
```

```
getch();
```

```
}
```

## Practical - 6

# Program on structure with initialization

Step 1: Declare the structure with initialization of variables.

Step 2: Call the declared structure with structure object.

Step 3: Print to the user to Enter the student details as roll no, name and percentage with following format specifier.

Step 4: Display the same to user.

Source Code:

```
#include < stdio.h >
#include < conio.h >
void main ()
{
    struct stud
    {
        char name [20];
        int roll;
        float per;
    };
    struct stud s1;
    clrscr();
}
```

Output :

042

Enter student details :

Enter roll no : 1732

Enter the name : Gaurav

Enter the fee : 90

Roll	Name	fee
1732	Gaurav	90.00000

```

printf("In Enter student details : ");
printf("In Enter rollno : ");
scanf("%d", &SI.roll);
printf("In Enter the name : ");
scanf("%S", &SI.name);
printf("Enter the percg");
scanf("%f", &SI.per);
printf("In't Roll\nName\nper");
printf("In't %d\n%S\n%f", SI.roll,
SI.name, SI.per);
getch();
}

```

#2.

### Algorithm:

- Step1: Start
- Step2: Declare structure student which will take input as roll number in integer, name in character & percentage in float
- Step3: Depending upon the number of inputs declare the structure objects
- Step4: Display to the user to enter roll, Name & Percentage for the 1<sup>st</sup> user.
- Step5: Display the same by scanning the inputs

## Source Code:

```

#include < stdio.h >
#include < conio.h >
void main ()
{
    struct student
    {
        int roll;
        char name[20];
        float per;
    } S1, S2;
    clrscr();
    printf ("\t Roll \t Name \t Percentage");
    scanf ("%d %s %f", &S1.roll, &S1.name,
           &S1.per);
    scanf ("%d %s %f", &S2.roll, &S2.name,
           &S2.per);
    printf ("In \t d \t %s \t %f", S1.roll, S1.name,
           S1.per);
    printf ("In \t d \t %s \t %f", S2.roll, S2.name,
           S2.per);
    getch ();
}

```

## Output :

(D)

- 1 Enter the ID: 1
- 1 Enter the name: Gaurav
- 1 Enter the address: Mumbai
- 2 Enter the ID: 2
- 2 Enter the ~~ID~~ Name: Daksh.
- 2 Enter the address: New York
- 3 Enter the ID: Rohit 3
- 3 Enter the Name: Rohit
- 3 Enter the address: Mumbai

Employee record is:

ID	Name	Address
1	Gaurav	Mumbai
2	Daksh	New York
3	Rohit	Mumbai

#3:

```

#include < stdio.h >
#include < conio.h >
void main()
{
    struct employee
    {
        int id;
        char name[30];
        char add[30];
    };
    struct employee e[50];
    int size, i;
    clrscr();
    printf("In Enter how many records you
           want to enter:");
    scanf("%d", &size);
    for (i = 1; i <= size; i++)
    {
        printf("In %d Enter the ID: ", i);
        scanf("%d", &e[i].id);
        printf("In %d Enter the name: ", i);
        scanf("%s", e[i].name);
        printf("In %d Enter the address: ", i);
        scanf("%s", e[i].add);
    }
    printf("In Employee record is: ");
    printf("In %d ID %s Name %s Address %s");
}

```

## Practical - 7

# 1 : Call by Value :

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

```
int x,y,z;
clrscr();
printf("Enter the value of X: ");
scanf("%d", &x);
printf("Enter the value of Y: ");
scanf("%d", &y);
z = x + y;
printf("In before function call the
number are: ");
printf("In x=%d & y=%d & z=%d",x,y,z);
z = Sample(x,y);
printf("After function call the number
are: ");
printf("In x=%d & y=%d & z=%d",x,y,z);
getch();
```

```
int Sample ( int a, int b )
```

```
int result;
```

$$a = b^2$$

$$b = a - 1$$

output :

048

Enter the value of X : 5

Enter the value of Y : 6

Before function call the numbers are

X = 5    Y = 6    z = 11

Inside the function

X = 10    Y = 20    z = 30

after function call the numbers are

X = 5    Y = 6    z = 30

```

printf ("Inside the function ");
printf ("In x=%d & y=%d & z=%d\n",
       a, b, result);
}

```

Algorithm:

- Step 1 : Start
- Step 2 : Declare function with integer parameters.
- Step 3 : Declare variables display the user Enter the value of x & y and scan the same.
- Step 4 : add the value and store in another variable
- Step 5 : Display the number before function call.
- Step 6 : Call the function and display the same.
- Step 7 : Define the declared function and print the same.
- Step 8 : Stop

```
#include <stdio.h>
#include <conio.h>
void main()
{
    char str[50];
    char st[10];
    clrscr();
    printf("In Enter a string : ");
    gets(str);
    printf("In Enter substring to find in
          the above string");
    gets(st);
    if (strstr(str, st) == NULL)
        printf ("In String Not Found!");
    else
    {
        printf ("In String found");
    }
    getch();
}
```

Enter a String : CS is all about<sup>048</sup>  
programming

Enter substring : all  
String found !