

## **ADDITIONAL EXERCISE**

### **TITLE:**

Write a program to declare two integer and one float variables then initialize them to 10, 15, and 12.6. Also print the variable values in the screen.

### **Objective:**

- ❖ To know different types of data types,
- ❖ To be familiar with printf function.

### **Problem analysis:**

Based on given problem, our program must initialize two integer variable to 10 and 15 and one float variable to 12.6.

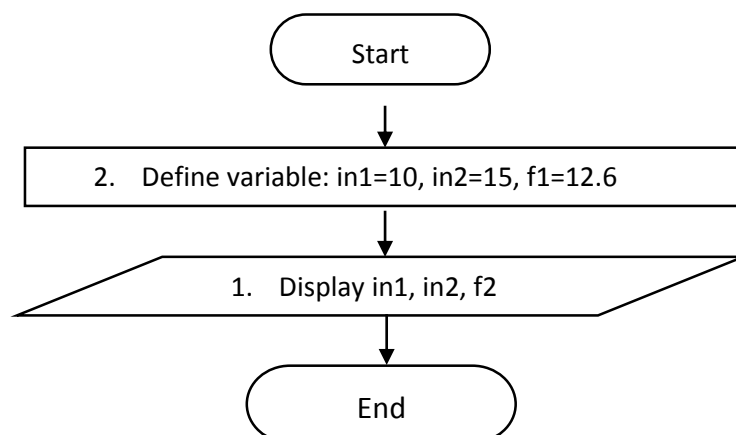
The given program should display above initialize variable.

Input variables	Necessary header files/functions/macros
in1,in2(int type) f1(float type)	stdio.h coino.h printf()

### **Algorithm:**

1. Start
2. Define variable: in1=10, in2=15, f1=12.6
3. Display in1, in2, f2
4. Stop

### **Flowchart:**



### **Code:**

```
#include<stdio.h>

#include<conio.h>

int main()
{
    clrscr();

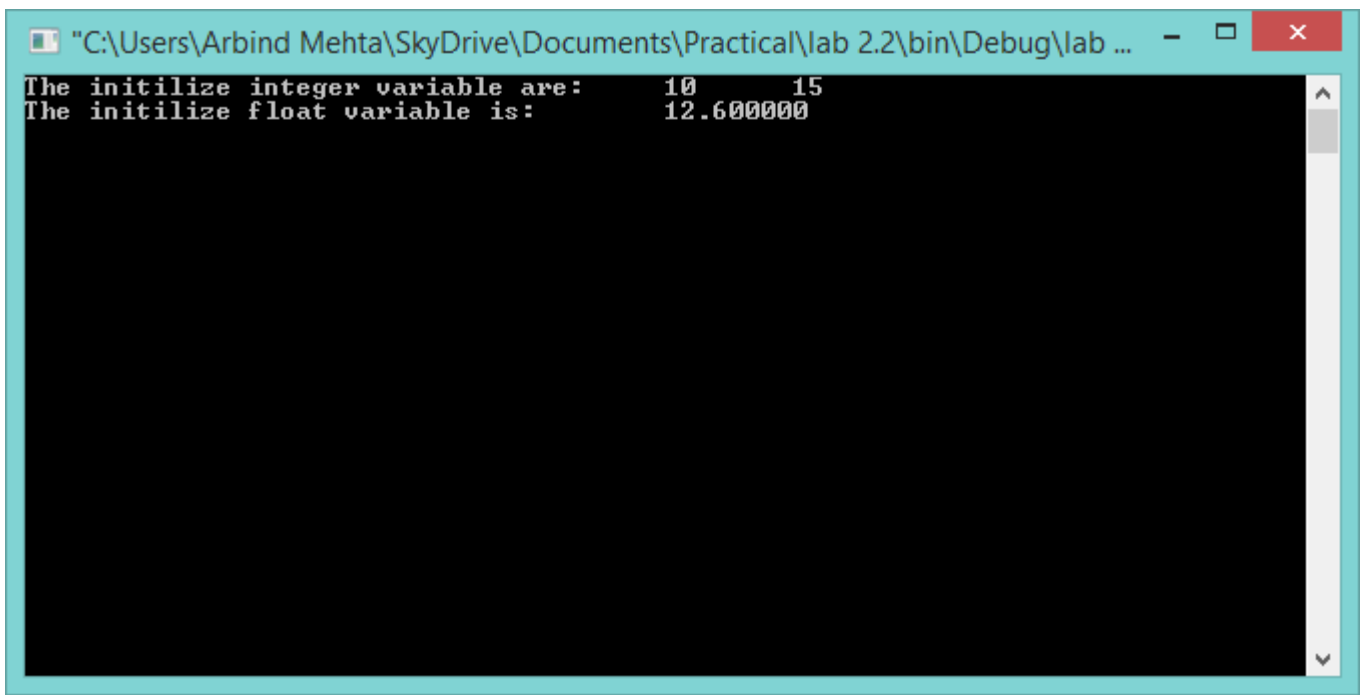
    int in1=10,in2=15;

    float f1=12.6;

    printf("The initialize integer variable are:\t%d\t%d\n",in1,in2);
    printf("The initialize float variable is:\t%f",f1);

    getch();
}
```

### **Output (Compilation, Debugging and Testing):**



```
"C:\Users\Arbind Mehta\SkyDrive\Documents\Practical\lab 2.2\bin\Debug\lab ... - □ ×
The initialize integer variable are: 10 15
The initialize float variable is: 12.600000
```

### **Discussion & Conclusion:**

In this lab of C programming, based on the focused objective(s) to understand about C data types and printf function.

### **Title:**

Write a C program to prompt the user to input 3 integer values and print these values in forward and reversed order.

### **Objective:**

- ❖ To be familiar with different argument of printf() and scanf() function.

### **Problem Analysis:**

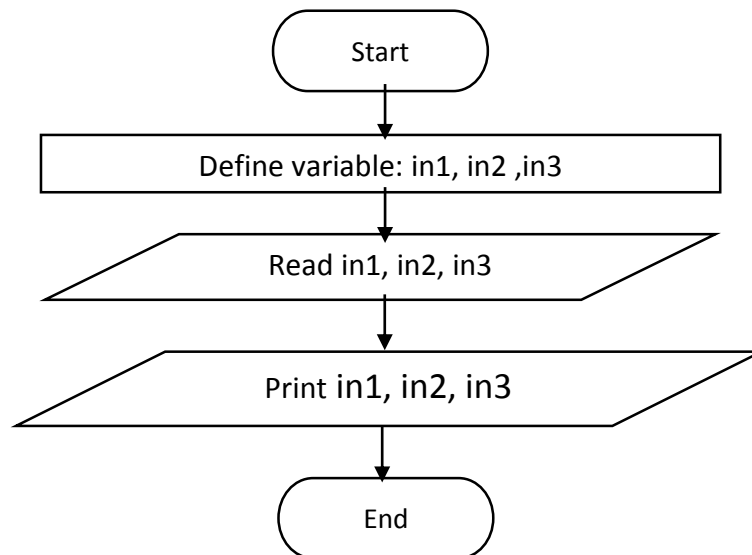
Based on problem, it is required to get the input of three integer value and display them in forward and reverse order.

Input variables	Necessary header files/functions/macros
in1, in2, in3(int type)	stdio.h coino.h scanf() printf()

### **Algorithm:**

1. Start
2. Define variables: n1, n2, n3
3. Take input from keyboard for all the input variables
4. Display the in1, in2, in2
5. Stop

## Flowchart:



## Code:

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

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

```
void main()
```

```
{
```

```
clrscr();
```

```
int n1,n2,n3;
```

```
printf("Enter three integer value:\n");
```

```
scanf("%d\t%d\t%d",&n1,&n2,&n3);
```

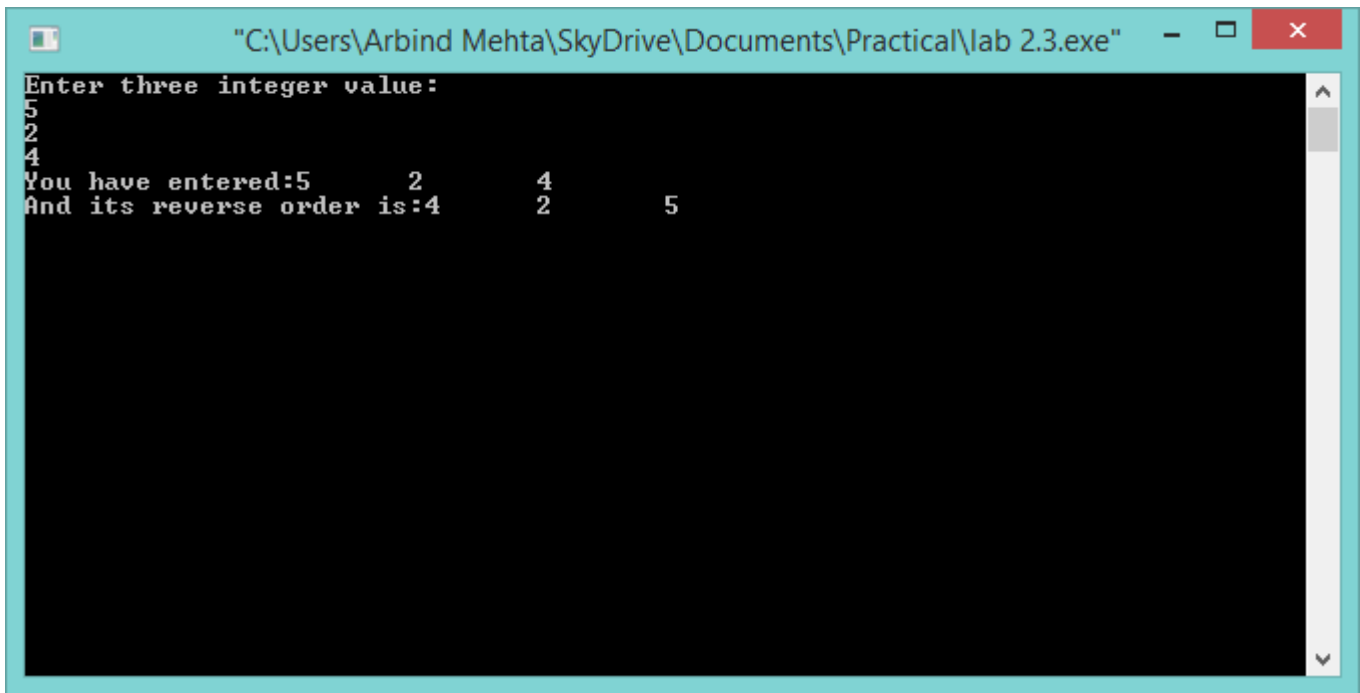
```
printf("You have entered:%d\t%d\t%d\n",n1,n2,n3);
```

```
printf("And its reverse order is:%d\t%d\t%d",n3,n2,n1);
```

```
getch();
```

```
}
```

## Output (Compilation, Debugging and Testing):



The screenshot shows a Windows command prompt window titled "C:\Users\Arbind Mehta\SkyDrive\Documents\Practical\lab 2.3.exe". The program prompts the user to "Enter three integer value:". The user enters the values 5, 2, and 4 on separate lines. The program then outputs "You have entered:5 2 4" and "And its reverse order is:4 2 5".

```
"C:\Users\Arbind Mehta\SkyDrive\Documents\Practical\lab 2.3.exe"
Enter three integer value:
5
2
4
You have entered:5 2 4
And its reverse order is:4 2 5
```

## Discussion & Conclusion:

In this lab of C programming, based on the focused objective(s) to learn the syntax and argument of printf() and scanf() function.

## Title:

Write a program to calculate simple and compound interest.

## Objective:

- ❖ To be familiar with different data types, operation and expression in C

## Problem Analysis:

Based on problem, it is required to find simple and compound interest by taking input of rate, time, principle amount and number of periods. The read float values are then passed into given expression to get simple and compound interest.

Simple interest= $((p*t*r)/100)$ ;

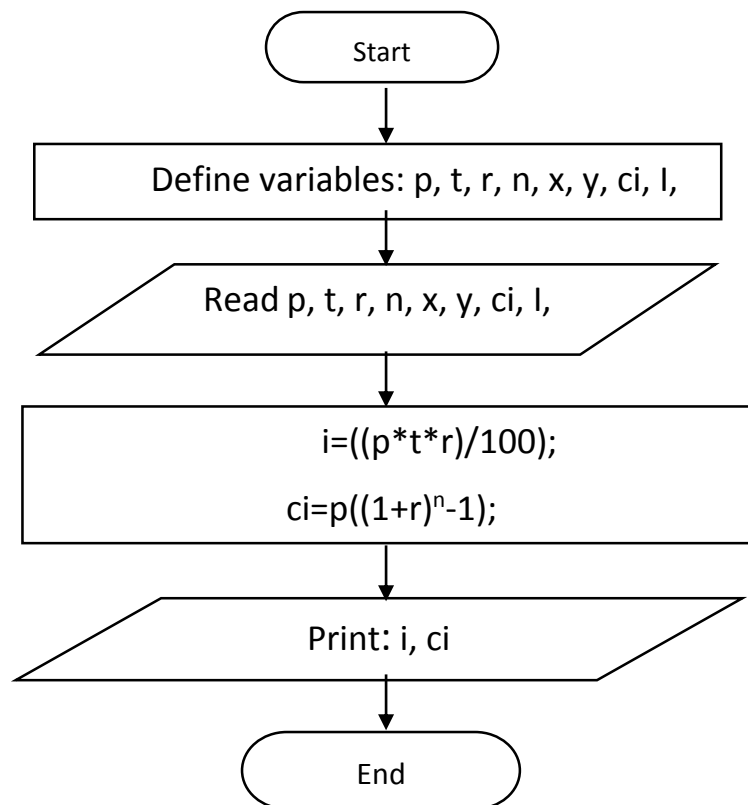
Compound interest= $p((1+r)^n-1)$ ;

Input variables	Processing variables/calculations	Output variables	Necessary header files/functions/macros
p, t, r, n(float type)	x, y(float type)	ci, i(float type)	stdio.h coino.h math.h scanf() printf()

### **Algorithm:**

1. Start
2. Define variables: p, t, r, n, x, y, ci, l,
3. Take input from keyboard for all the input variables
4. Calculate simple and compound interest :
5.  $i = ((p * t * r) / 100);$
6.  $ci = p((1 + r)^n - 1);$
7. Display the i, ci
8. Stop

### **Flowchart:**



### **Code:**

```
#include<stdio.h>

#include<conio.h>

#include<math.h>

void main()
{
    clrscr();

    float t,r,p,ci,i,x,y,n;

    printf("Enter pinciple, time, rate, number of period:\n");
    scanf("%f%f%f%f",&p,&t,&r,&n);

    i=((p*t*r)/100);
    x=pow((1+r),n);
    ci=(p*(x-1));
    printf("The interest is:%f\n",i);
    printf("The compoud interest is:%f",ci);

}
```

## Output (Compilation, Debugging and Testing):

```
"C:\Users\Arbind Mehta\SkyDrive\Documents\Practical\lab 2.4.exe"
Enter pinciple, time, rate, number of period:
1000
5
2
3
The interest is:100.000000
The compoud interest is:26000.000000
Process returned 36 (0x24) execution time : 8.697 s
Press any key to continue.
```

## Discussion & Conclusion:

In this lab of C programming, based on the focused objective(s) to learn different data types, operation and expression in C

## Title:

Write a program to swap two variables values with and without using third variables

## Objective:

- ❖ To be familiar with different data types, operation and expression in C

## Problem Analysis:

Based on problem, it is required to define two variable and assingn some value to it. Then a third variable of same data type is define. Then the value of one variable is copied to other and hence swaping is done.



Input variables	Processing variables/calculations	Necessary header files/functions/macros
a1, a2(int type)	c(int type)	stdio.h coino.h printf()

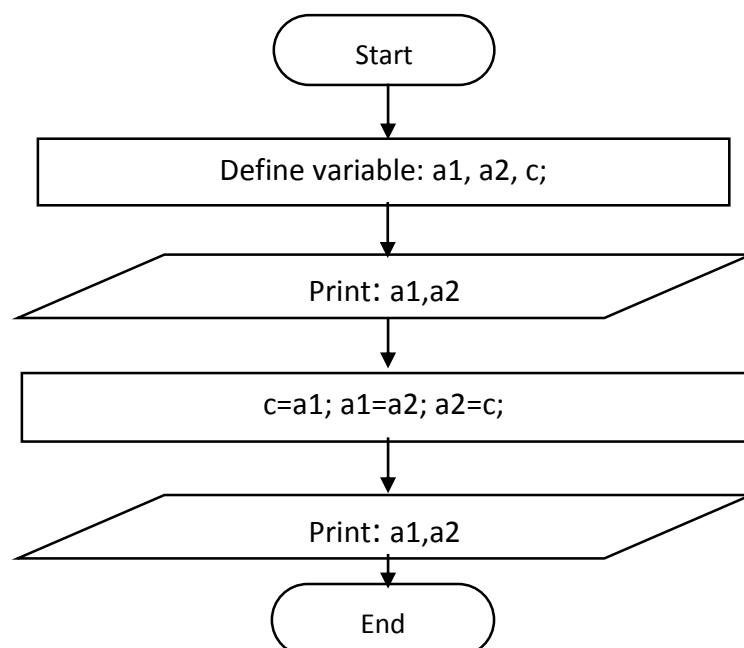
### **Algorithm:**

1. Start
2. Define variables: a1, a2, c
3. Display the a1, a2.
4. c=a1;
5. a1=a2;
6. a2=c
7. Display the a1, a2.
8. Stop

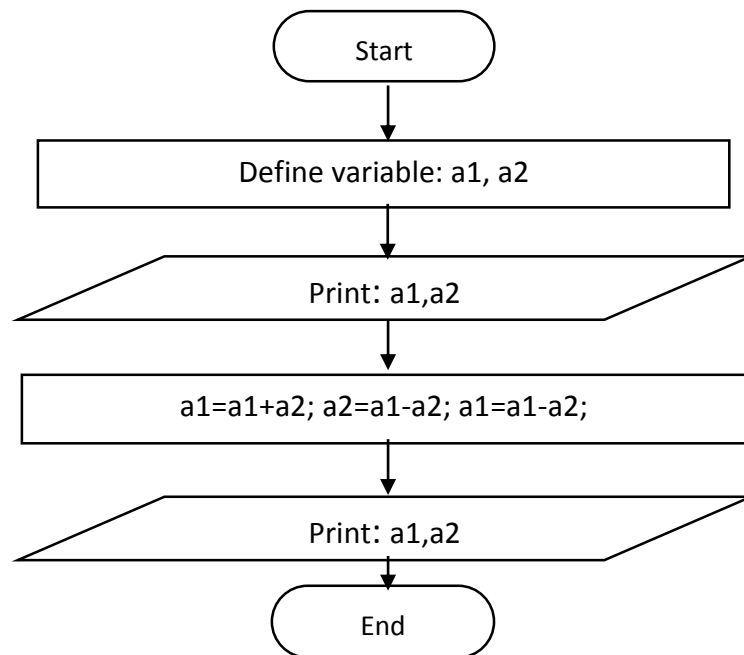
### **Algorithm(without using third variable):**

1. Start
2. Define variables: a1, a2,
3. Display the a1, a2.
4. a1=a1+a2;
5. a2=a1-a2;
6. a1=a1-a2;
7. Display the a1, a2.
8. Stop

### **Flowchart:**



### Flowchart(without using third variable):



### Code:

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

```
void main()
```

```
{
```

```
clrscr();
```

```
int a1=1,a2=2,c;
```

```
printf("a and b are:%d,%d\n",a1,a2);
```

```
c=a2;
```

```
a2=a1;
```

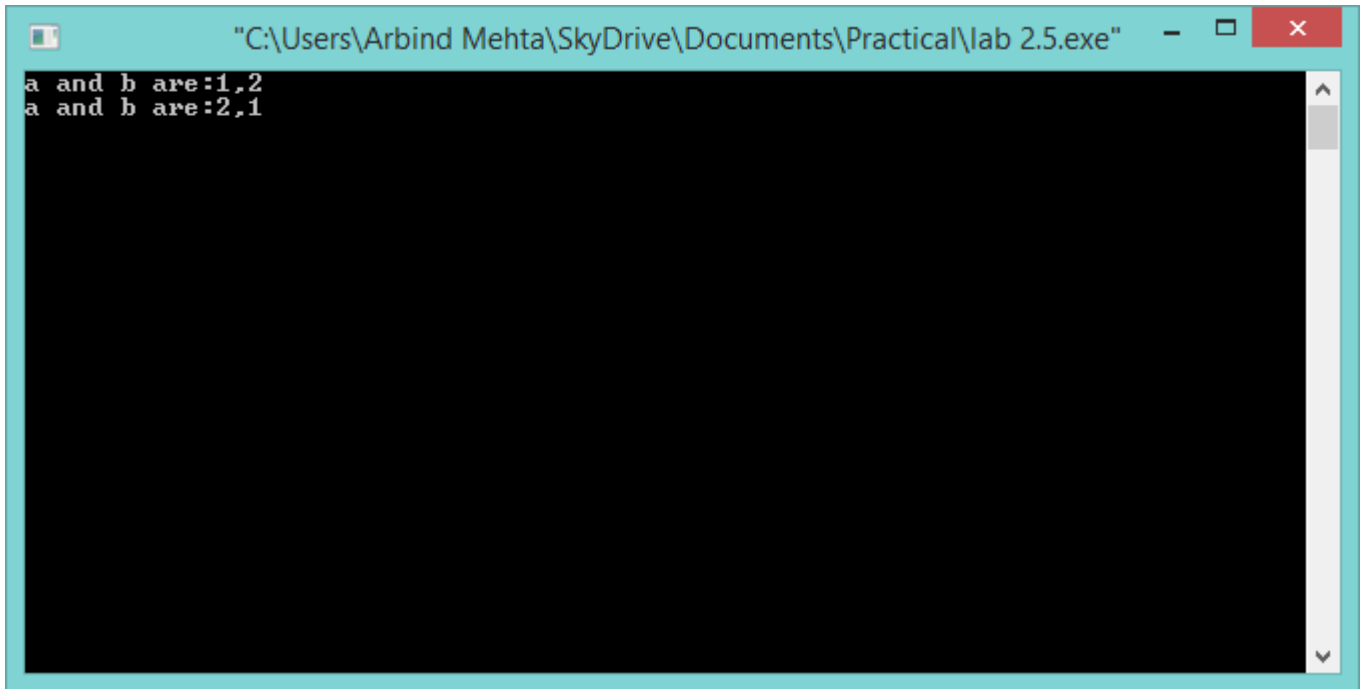
```
a1=c;
```

```
printf("a and b are:%d,%d",a1,a2);
```

```
getch();
```

```
}
```

### **Output (Compilation, Debugging and Testing):**



```
"C:\Users\Arbind Mehta\SkyDrive\Documents\Practical\lab 2.5.exe"
a and b are:1,2
a and b are:2,1
```

### **Code(without using third variable):**

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

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

```
void main()
```

```
{
```

```
clrscr();
```

```
int a=1,b=2;
```

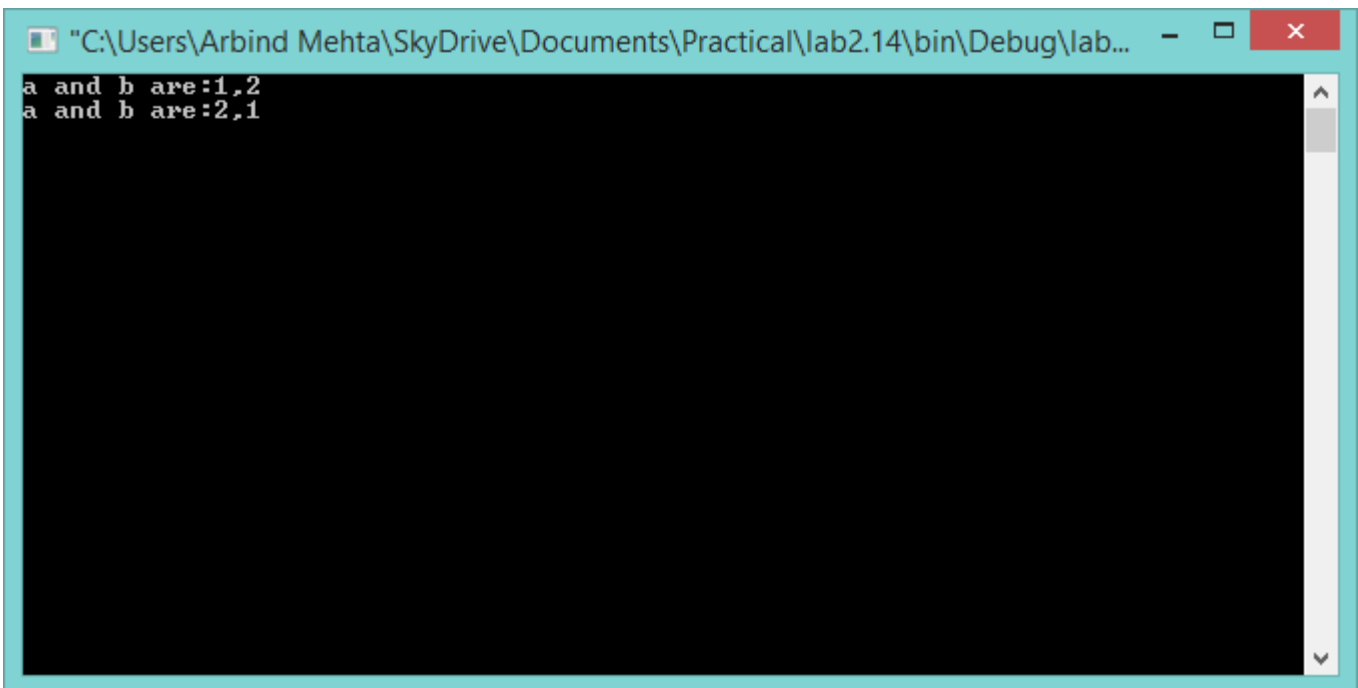
```
printf("a and b are:%d,%d\n",a,b);
```

```
a=a+b;
```

```
b=(a-b);
```

```
a=(a-b);  
printf("a and b are:%d,%d",a,b);  
  
getch();  
  
}
```

### **Output (Compilation, Debugging and Testing):**



```
"C:\Users\Arbind Mehta\SkyDrive\Documents\Practical\lab2.14\bin\Debug\lab...  
a and b are:1,2  
a and b are:2,1
```

### **Discussion & Conclusion:**

In this lab of C programming, based on the focused objective(s) to learn different data types, operation and expression in C

### **Title:**

Write a program to check odd or even number (a) using modulus operator (b) using bitwise operator (c) without using bitwise and modulus operator (d) using conditional operator.

### **Objective:**

- ❖ To be familiar with different data types, operation and expression in C

### **Problem Analysis:**

Based on problem, it is required to get the input of integer number and check whether the given number is even or odd.

Input variables	Processing variables/calculations	Necessary header files/functions/macros
n(int type)	ans(int type)	stdio.h conio.h scanf() printf()

### **Algorithm(using modulus operator):**

1. Start
2. Define variables: n
3. Take input from keyboard for all the input variables
4. If n is divisible by 2, display n is even number else display n is odd number
5. Stop

### **Algorithm(using conditional operator):**

1. Start
2. Define variables: n, ans
3. Take input from keyboard for all the input variables
4. If n is divisible by 2, display n is even number else display n is odd number
5. Stop

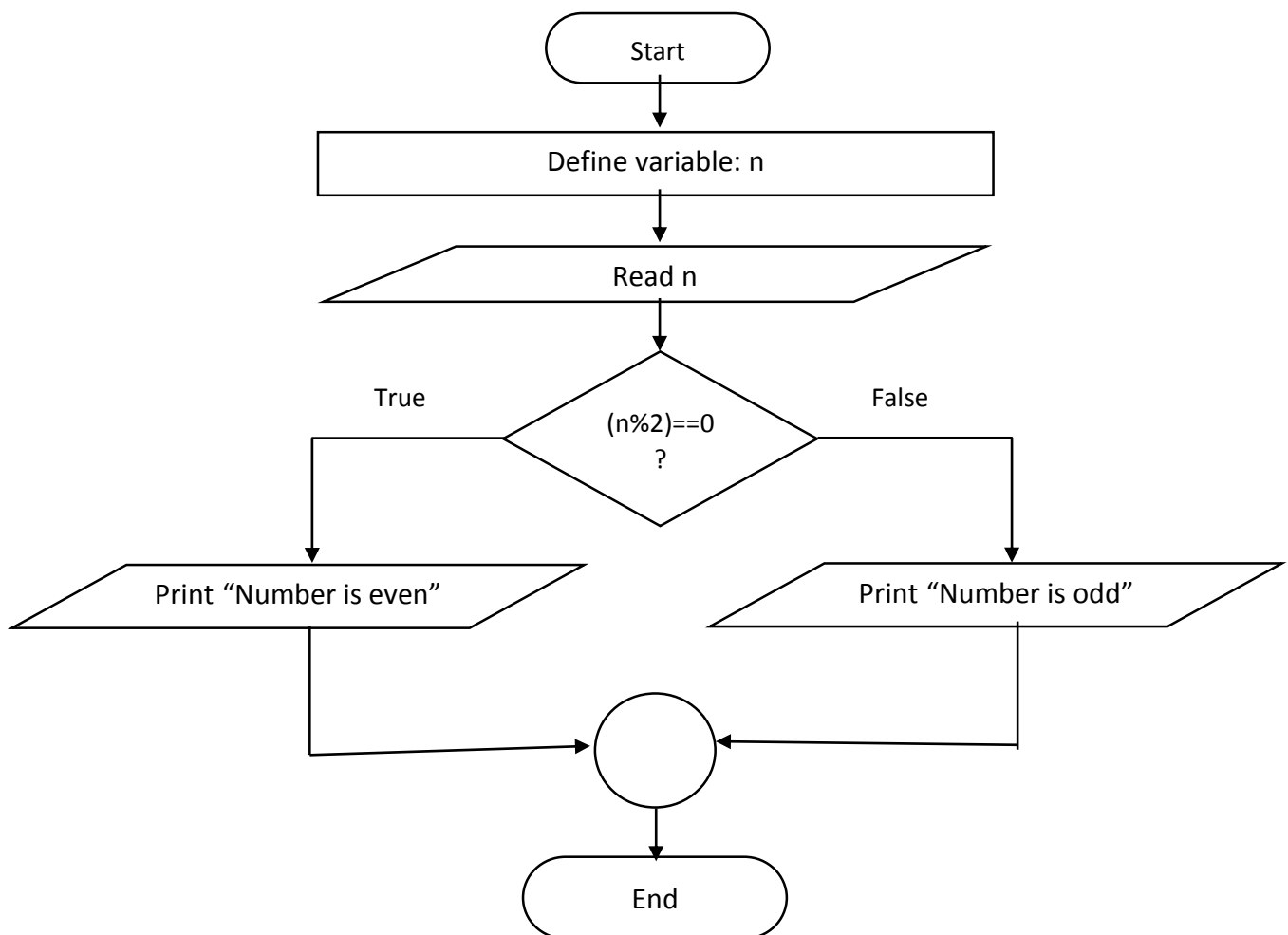
### **Algorithm(using bitwise operator):**

1. Start
2. Define variables: n
3. Take input from keyboard for all the input variables
4. If  $((n \& 1) = 1)$  print odd else print even
5. Stop

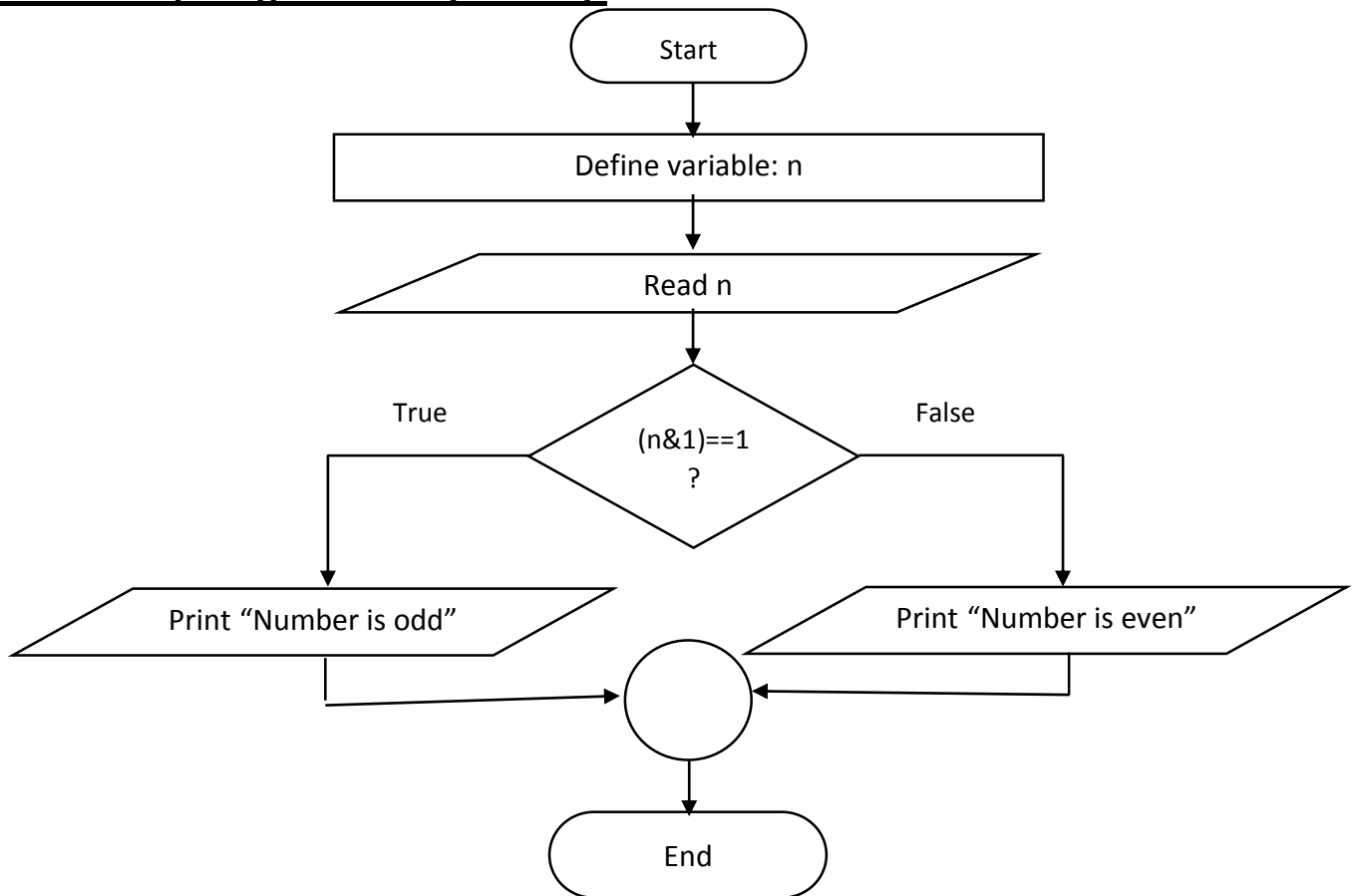
### Algorithm(without using modulus and bitwise operator):

1. Start
2. Define variables: n
3. Take input from keyboard for all the input variables
4. If twice of half of given number is same number, print the number is even else print number is odd.
5. Stop

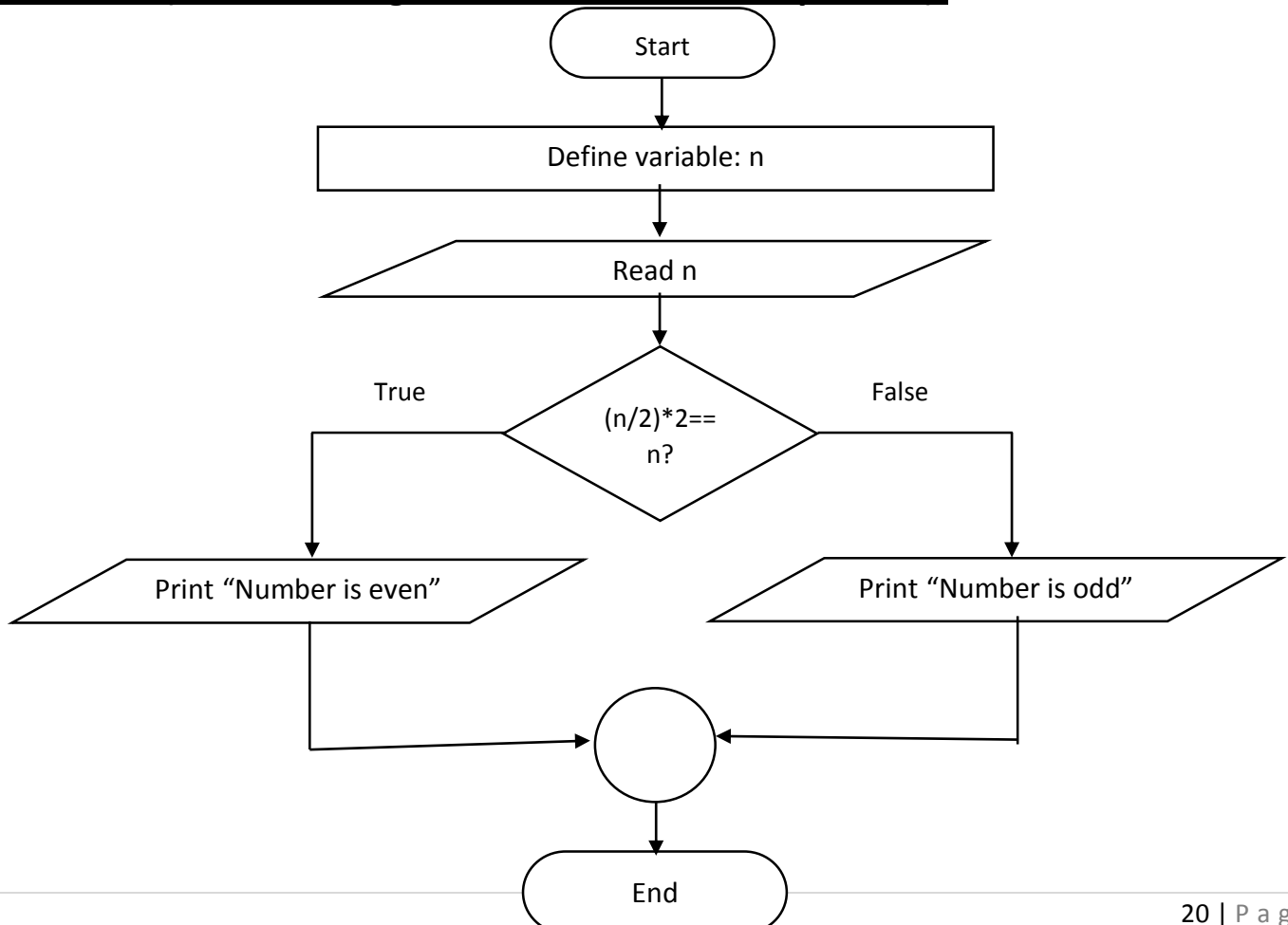
### Flowchart(using modulus operator/using conditional operator):



### Flowchart(using bitwise operator):



### Flowchart(without using modulus and bitwise operator):



### **Code(using modulus operator):**

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

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

```
int main()
```

```
{
```

```
clrscr();
```

```
int n;
```

```
printf("Enter the number to be checked:");
```

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

```
if((n%2)==0)           //using modulus operator.
```

```
printf("%d is even number.",n);
```

```
else
```

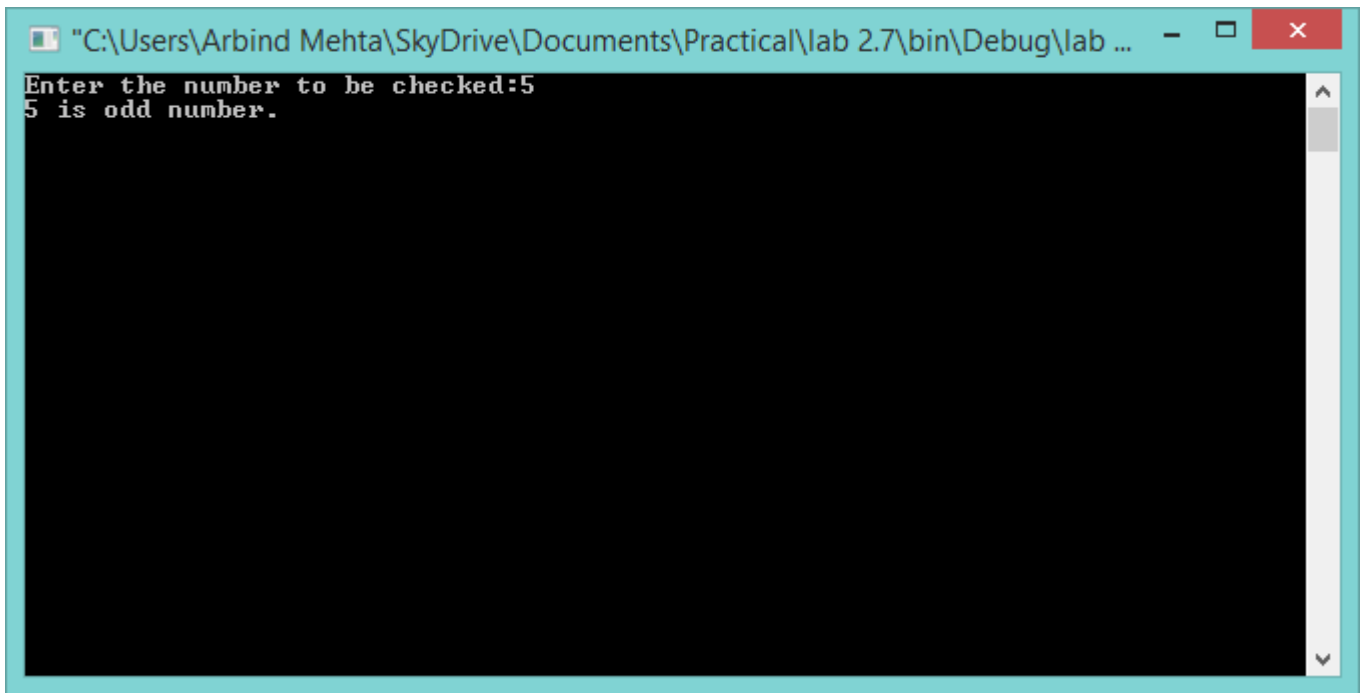
```
printf("%d is odd number.",n);
```

```
getch();
```

```
}
```



## Output (Compilation, Debugging and Testing):



```
"C:\Users\Arbind Mehta\SkyDrive\Documents\Practical\lab 2.7\bin\Debug\lab ... - □ ×  
Enter the number to be checked:5  
5 is odd number.
```

## Code(using conditional operator):

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

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

```
void main()
```

```
{
```

```
clrscr();
```

```
int n,ans;
```

```
printf("Enter the number to be checked:");
```

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

```
ans=((n%2)==0) ? 1 : 0; //using conditional operator
```

```
if(ans==1)
```

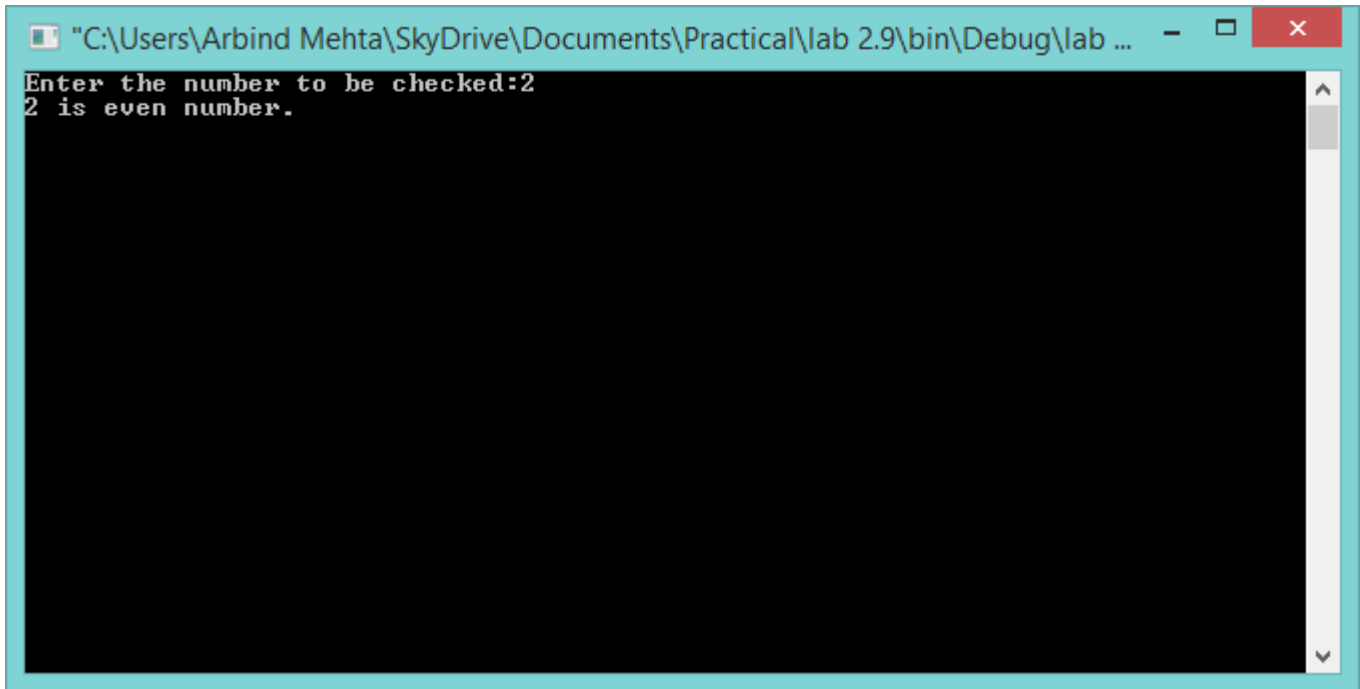
```
printf("%d is even number.\n",n);
```

```
else
```

```
printf("%d is odd number.",n);
```

```
    getch();  
  
}
```

### **Output (Compilation, Debugging and Testing):**

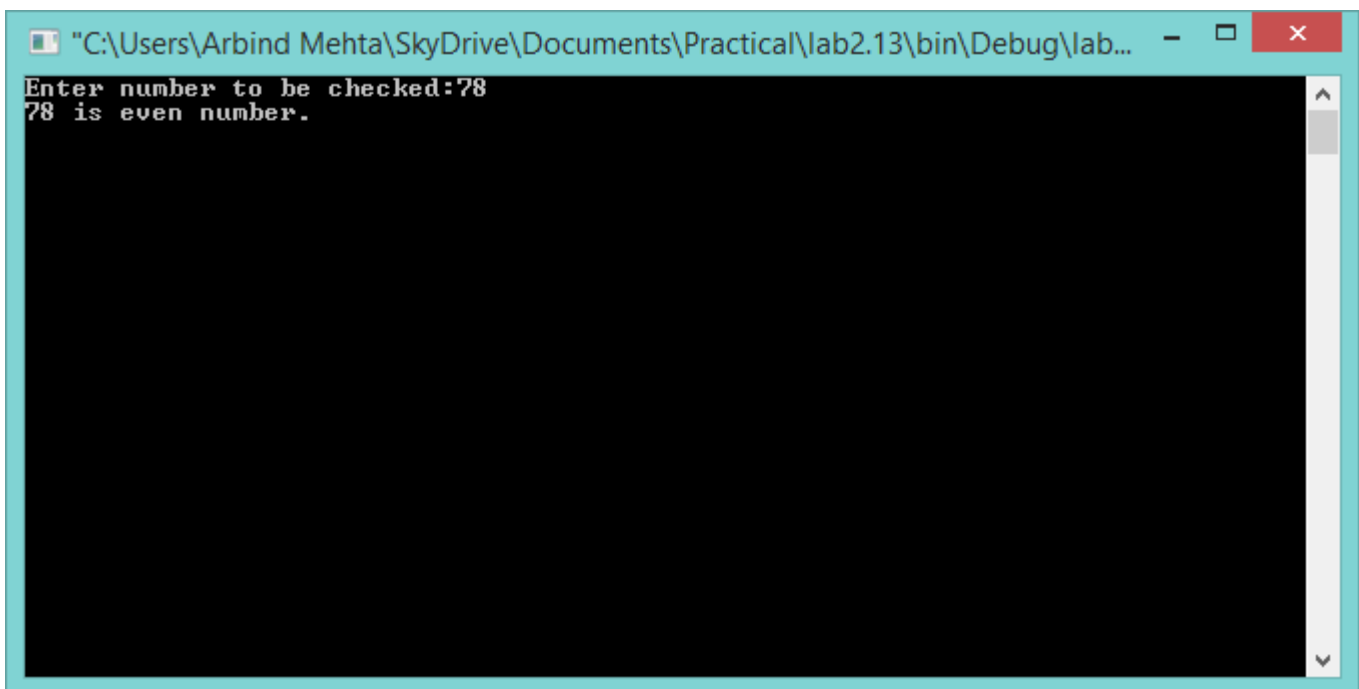


### **Coding(using bitwise operator):**

```
#include <stdio.h>  
  
#include <conio.h>  
  
int main()  
{  
    clrscr();  
    int n;  
  
    printf("Enter number to be checked:");  
    scanf("%d",&n);
```

```
if((n&1)==1)
printf("%d is odd number.",n);
else
printf("%d is even number.",n);
getch();
}
```

### **Output (Compilation, Debugging and Testing):**



```
"C:\Users\Arbind Mehta\SkyDrive\Documents\Practical\lab2.13\bin\Debug\lab... - □ ×
Enter number to be checked:78
78 is even number.
```

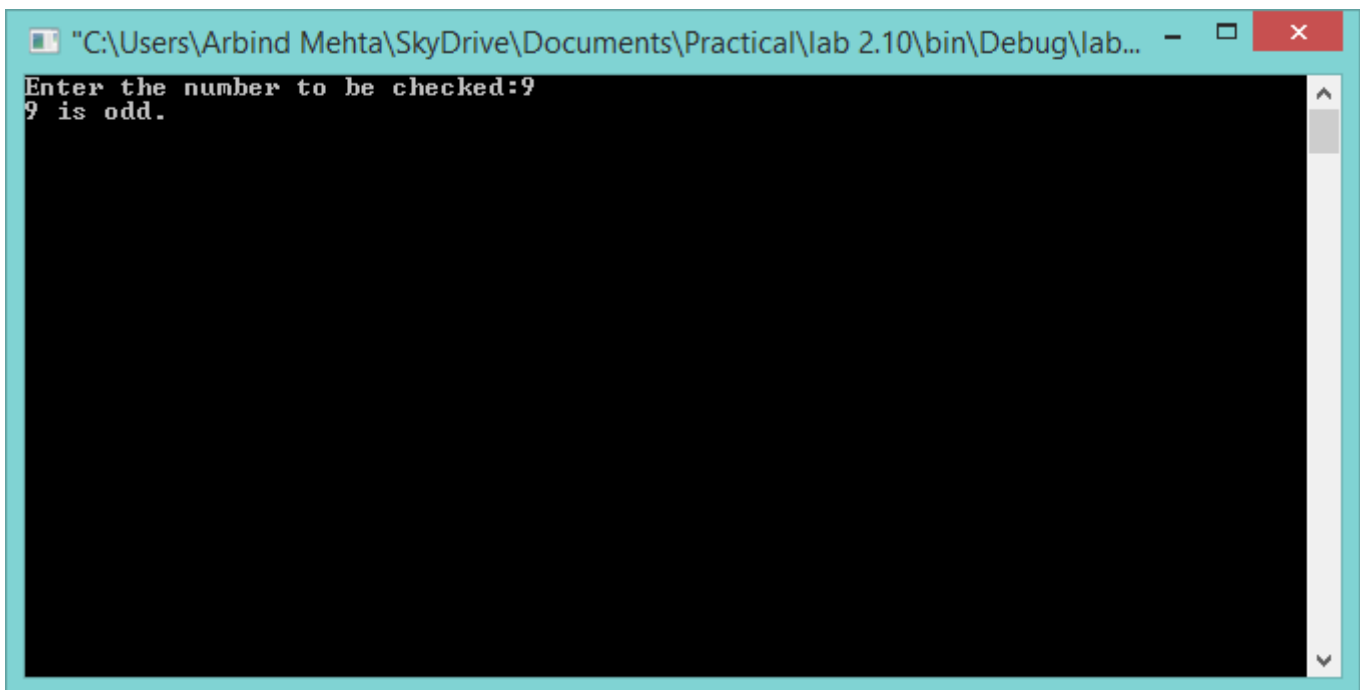
### **Code(without using modulus and bitwise operator):**

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

void main()
{ clrscr();
  int n,ans;
```

```
printf("Enter the number to be checked:");  
scanf("%d",&n);  
  
if(((n/2)*2)==n)  
printf("%d is even number.",n);  
else  
printf("%d is odd.",n);  
getch();  
}
```

### **Output (Compilation, Debugging and Testing):**



```
"C:\Users\Arbind Mehta\SkyDrive\Documents\Practical\lab 2.10\bin\Debug\lab... - □ ×  
Enter the number to be checked:9  
9 is odd.
```

### **Discussion & Conclusion:**

In this lab of C programming, based on the focused objective(s) to learn different data types, modulus, bitwise and conditional operator and expression in C

## Title:

Print the value of y for given x=2 & z=4 and analyze the output. a.  $y = x++ + ++x$ ; b.  $y = ++x + ++x$ ; c.  $y = ++x + ++x + ++x$ ; d.  $y = x > z$ ; e.  $y = x > z ? x : z$ ; f.  $y = x \& z$ ; g.  $y = x >> 2 + z << 1$ .

## Objective:

- ❖ To be familiar with different data types, operation and expression in C

## Problem Analysis:

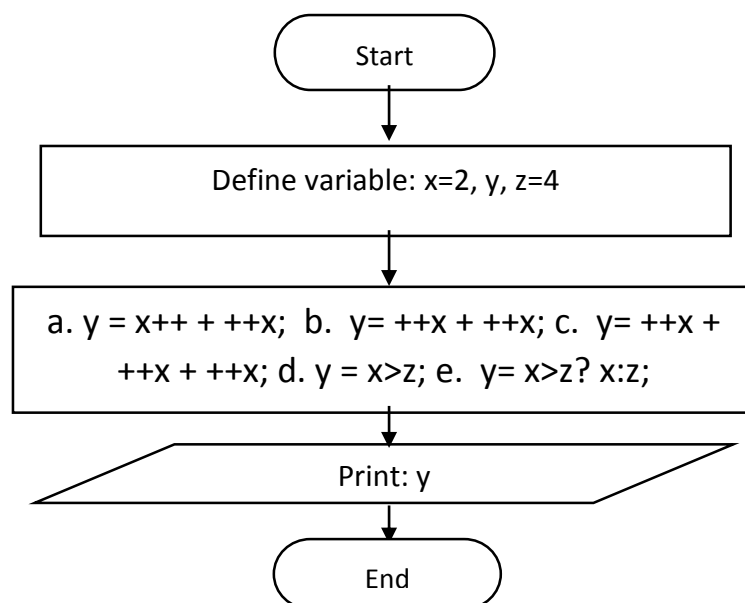
Based on problem, it is required to define three integer variable and assign the value to them

Input variables	Processing variables/calculations	Output variables	Necessary header files/functions/macros
x, z(int type)	y(int type)	y(int type)	stdio.h coino.h printf()

## Algorithm:

1. Start
2. Define variables: x=2, y, z=4.
3. a.  $y = x++ + ++x$ ; b.  $y = ++x + ++x$ ; c.  $y = ++x + ++x + ++x$ ; d.  $y = x > z$ ; e.  $y = x > z ? x : z$ ;  
a. f.  $y = x \& z$ ; g.  $y = x >> 2 + z << 1$ .
4. Display the y
5. Stop

## Flowchart:



## Code:

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

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

```
int main()
```

```
{
```

```
    clrscr();
```

```
    int x=2,y,z=4;
```

```
    y=(x++)+(++x);
```

```
    printf("y=%d\n",y);
```

```
    y=(++x)+(++x);
```

```
    printf("y=%d\n",y);
```

```
    y=(++x)+(++x)+(++x);
```

```
    printf("y=%d\n",y);
```

```
    y=x>z;
```

```
    printf("y=%d\n",y);
```

```
    y=x>z?x:z;
```

```
    printf("y=%d\n",y);
```

```
    y=x&z;
```

```
    printf("y=%d\n",y);
```

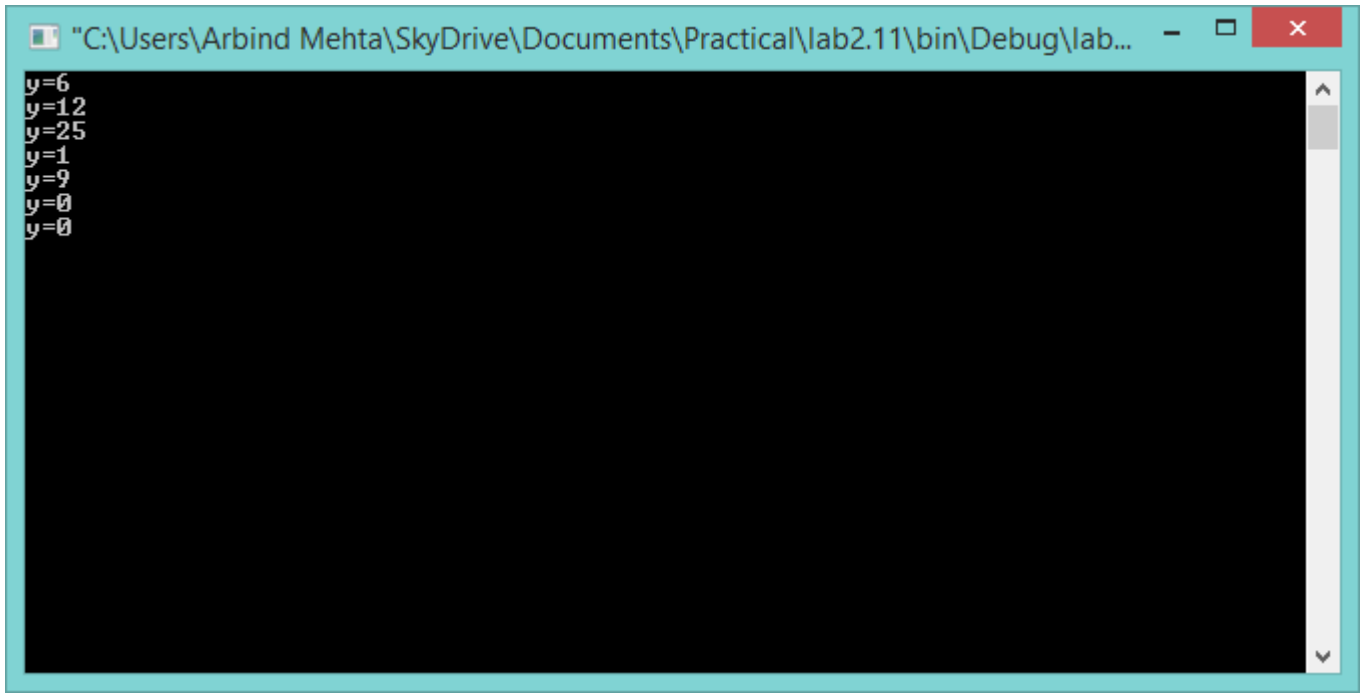
```
    y=x>>2+z<<1;
```

```
printf("y=%d\n",y);

getch();

}
```

### **Output (Compilation, Debugging and Testing):**



```
"C:\Users\Arbind Mehta\SkyDrive\Documents\Practical\lab2.11\bin\Debug\lab...  
y=6  
y=12  
y=25  
y=1  
y=9  
y=0  
y=0
```

### **Discussion & Conclusion:**

In this lab of C programming, based on the focused objective(s) to learn different data types, operation and expression in C

### **Title:**

Write a program to print the size of char, float, double and long double data types in C

### **Objective:**

- ❖ To be familiar with sizeof() function in C

## **Problem Analysis:**

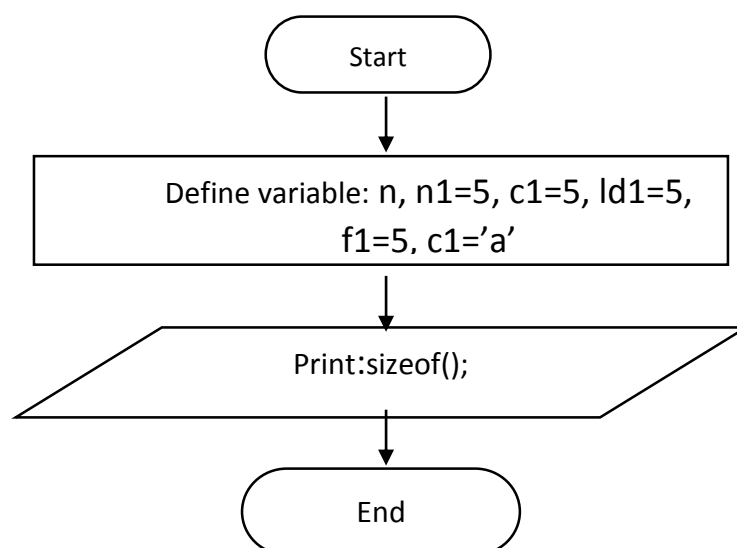
Based on problem, it is required to declare five variable of type integer, character, float, double, long double. Then by using sizeof() function return the size of variable which is been passed as argument this function.

Input variables	Necessary header files/functions/macros
n1(int type),d1(double type), c1(char type), f1(float type), ld1(long double type)	stdio.h coino.h printf()

## **Algorithm:**

1. Start
2. Define variables:n, n1=5, c1=5, ld1=5, f1=5, c1='a'
3. Display sizeof();
4. Stop

## **Flowchart:**





**Code:**

```
#include <stdio.h>

#include <stdlib.h>
```

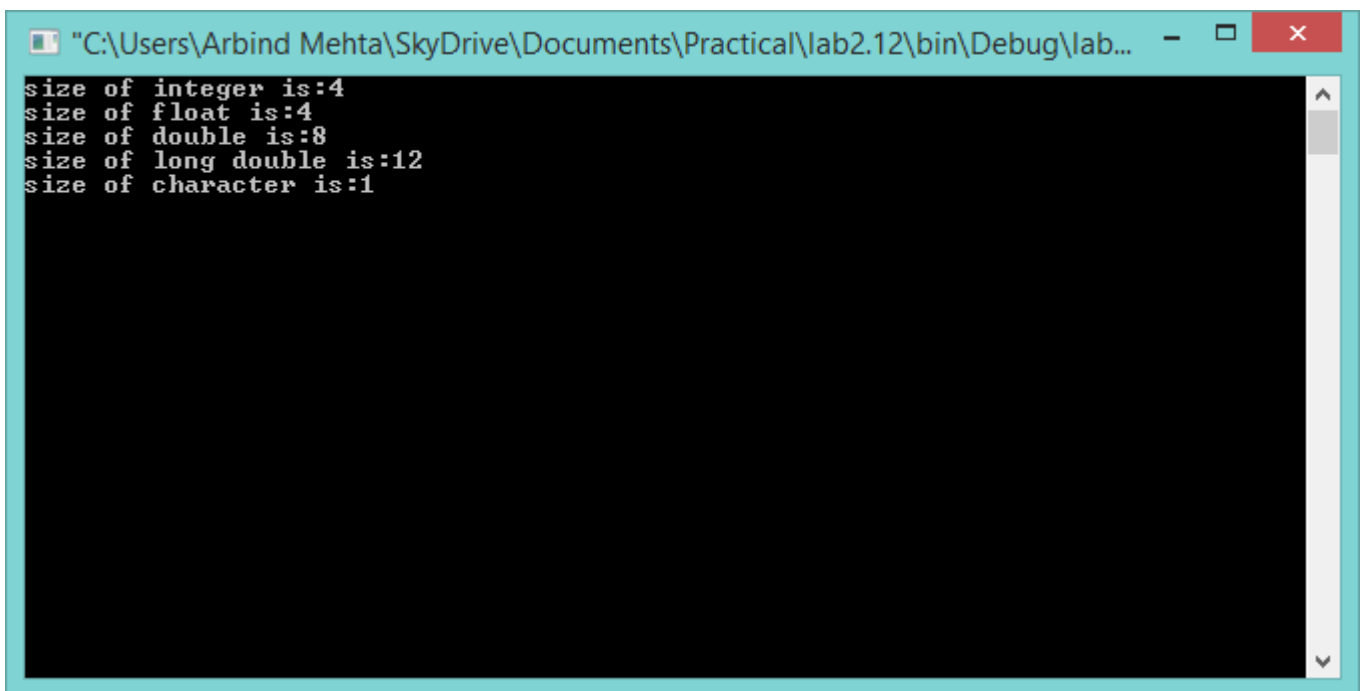
```
int main()
{
    clrscr();
    int n1=5;
    float f1=5;
    double d1=5;
    long double ld1=5;
    char c1='a';

    printf("size of integer is:%d\n",sizeof(n1));
    printf("size of float is:%d\n",sizeof(f1));
    printf("size of double is:%d\n",sizeof(d1));
    printf("size of long double is:%d\n",sizeof(ld1));
    printf("size of character is:%d\n",sizeof(c1));

    getch();

}
```

## **Output (Compilation, Debugging and Testing):**

A screenshot of a Windows command prompt window. The title bar shows the file path: "C:\Users\Arbind Mehta\SkyDrive\Documents\Practical\lab2.12\bin\Debug\lab...". The command prompt has a black background with white text. The output displayed is:

```
size of integer is:4  
size of float is:4  
size of double is:8  
size of long double is:12  
size of character is:1
```

## **Discussion & Conclusion:**

In this lab of C programming, based on the focused objective to be familiar with sizeof() function, its syntax, argument and its uses.