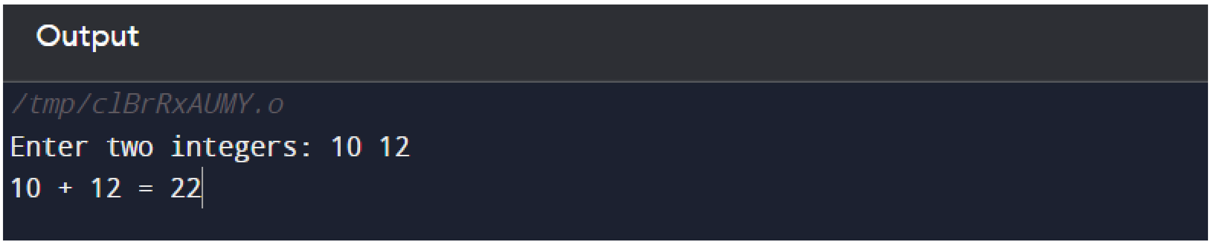
OOPS C++

. Write A Program In C++ To Add To Two Numbers through The Keyboard.

```
#include <iostream>
using namespace std;

int main() {

  int first_number, second_number, sum;

  cout << "Enter two integers: ";
  cin >> first_number >> second_number;

  // sum of two numbers in stored in variable sumOfTwoNumbers
  sum = first_number + second_number;

  // prints sum
  cout << first_number << " + " <<  second_number << " = " << sum;

  return 0;
}
```

OUTPUT :

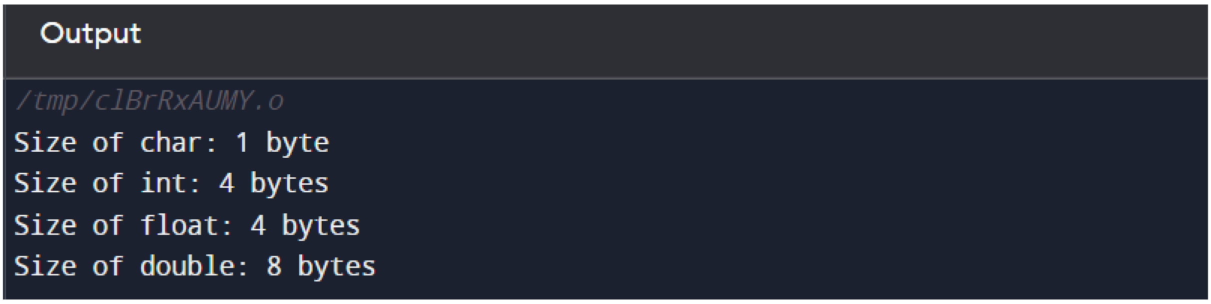


```
/tmp/clBrRxAUMY.o
Enter two integers: 10 12
10 + 12 = 22
```

. Write C++ Program To Find Size Of Data Type.

```
#include <iostream>
using namespace std;

int main()
{
   cout << "Size of char: " << sizeof(char) << " byte" << endl;
   cout << "Size of int: " << sizeof(int) << " bytes" << endl;
   cout << "Size of float: " << sizeof(float) << " bytes" << endl;
   cout << "Size of double: " << sizeof(double) << " bytes" << endl;

   return 0;
}
```

OUTPUT :



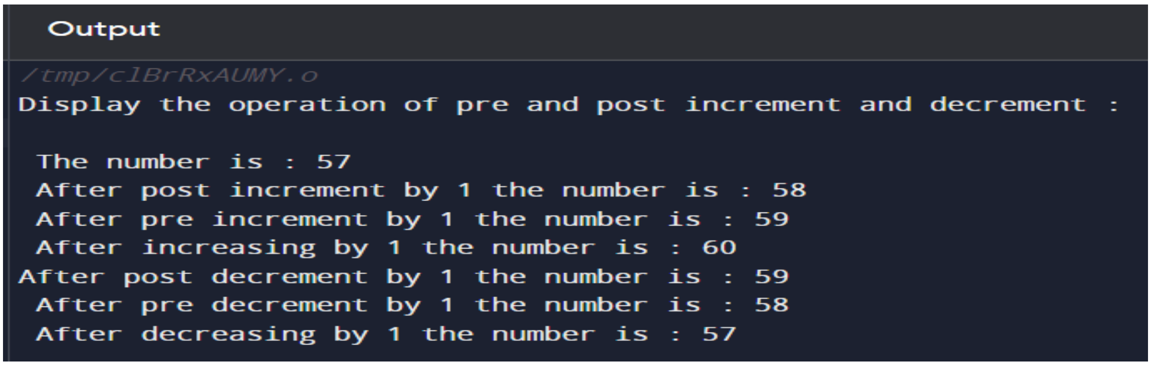
```
/tmp/clBrRxAUMY.o
Size of char: 1 byte
Size of int: 4 bytes
Size of float: 4 bytes
Size of double: 8 bytes
```

.Write A Program In C++ To Display The Operation Of Pre And Post Increment And Decrement.

```
#include <iostream>
using namespace std;
```

```
int main()
{
   int num = 57;
    cout << "Display the operation of pre and post increment and decrement :\n\n";
   cout <<" The number is : " << num << endl;
   num++;
   cout <<" After post increment by 1 the number is : " << num << endl;
   ++num;
   cout <<" After pre increment by 1 the number is : " << num << endl;
   num = num + 1;
   cout <<" After increasing by 1 the number is : " << num << endl;
   num--;
   cout <<" After post decrement by 1 the number is : " << num << endl;
   --num;
   cout <<" After pre decrement by 1 the number is : " << num << endl;
   num = num - 1;
   cout <<" After decreasing by 1 the number is : " << num << endl;
   cout << endl;
   return 0;
}
```

OUTPUT :

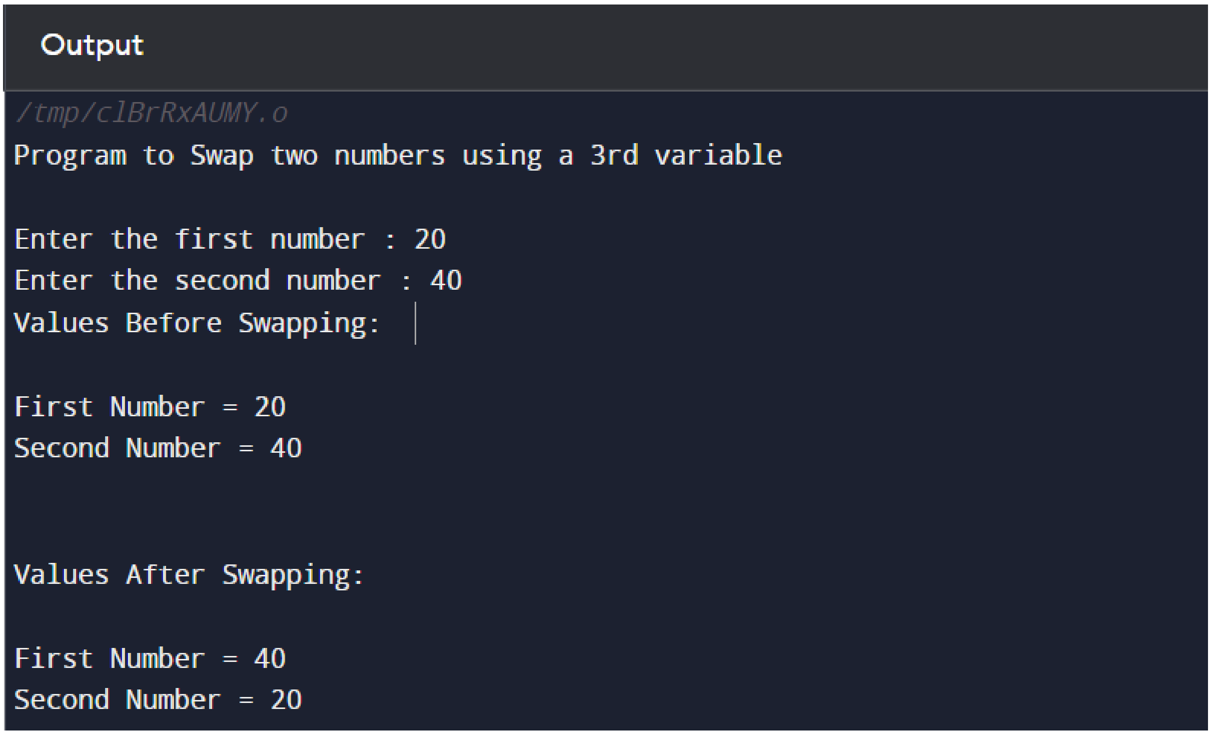


```
/tmp/clBrRxAUMY.o
Display the operation of pre and post increment and decrement :

 The number is : 57
 After post increment by 1 the number is : 58
 After pre increment by 1 the number is : 59
 After increasing by 1 the number is : 60
After post decrement by 1 the number is : 59
 After pre decrement by 1 the number is : 58
 After decreasing by 1 the number is : 57
```

. Write A Program In C++ To Swap Two Numbers Using Third Variable.

```
#include <iostream>
using namespace std;

int main()
{
  cout << "Program to Swap two numbers using a 3rd variable\n\n";

   int a,b,temp;

   cout << "Enter the first number : ";
   cin >> a;

   cout << "Enter the second number : ";
   cin >> b;

   cout<<"\n\nValues Before Swapping:  \n"<<endl;
   cout<<"First Number = " << a <<endl;
   cout<<"Second Number = " << b <<endl;

   temp = a;
   a = b;
   b = temp;
   cout << "\n\nValues After Swapping:  \n"<<endl;
   cout << "First Number = " << a <<endl;
   cout << "Second Number = " << b <<endl;
   cout << "\n\n\n";
           return 0;}
```

OUTPUT :



Output

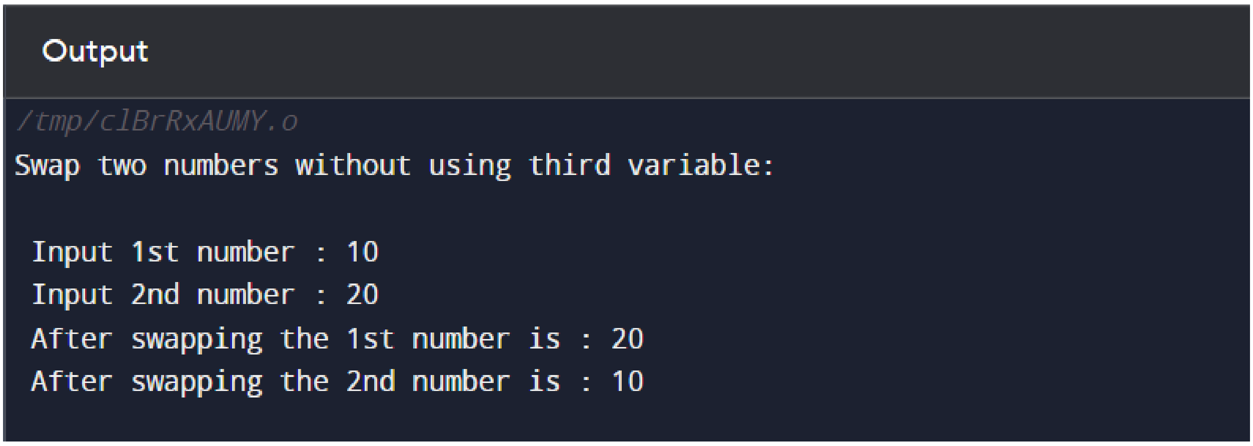
```
/tmp/clBrRxAUMY.o
Program to Swap two numbers using a 3rd variable

Enter the first number : 20
Enter the second number : 40
Values Before Swapping: 

First Number = 20
Second Number = 40



Values After Swapping:

First Number = 40
Second Number = 20
```

. Write A Program In C++ Which Swap The Value Of Two Variable Not Using The Third Variable.

```
#include <iostream>
using namespace std;

int main()
{
   cout << "Swap two numbers without using third variable:\n\n";

   int num1, num2, temp;
   cout << " Input 1st number : ";
   cin >> num1 ;
   cout << " Input 2nd number : ";
   cin >> num2;
```

```
  num2=num2+num1;
  num1=num2-num1;
  num2=num2-num1;
   cout << " After swapping the 1st number is : "<< num1 <<"\n" ;
   cout << " After swapping the 2nd number is : "<< num2 <<"\n\n" ;
}
```

OUTPUT :



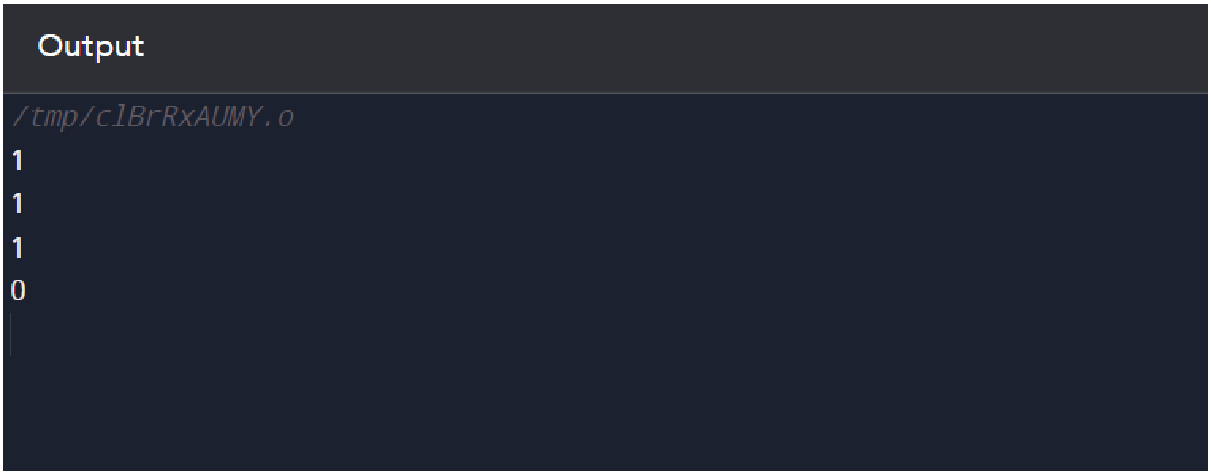
```
/tmp/clBrRxAUMY.o
Swap two numbers without using third variable:

 Input 1st number : 10
 Input 2nd number : 20
 After swapping the 1st number is : 20
 After swapping the 2nd number is : 10
```

. Write A C++ Program To Check If A Given Positive Number Is A Multiple Of 3 Or 7.

```
#include <iostream>
using namespace std;

bool test(int n)
    {
       return n % 3 == 0 || n % 7 == 0;
    }

int main()
```

```
{
 cout << test(3) << endl;
 cout << test(14) << endl;
 cout << test(12) << endl;
 cout << test(37) << endl;
 return 0;
}
```

OUTPUT :



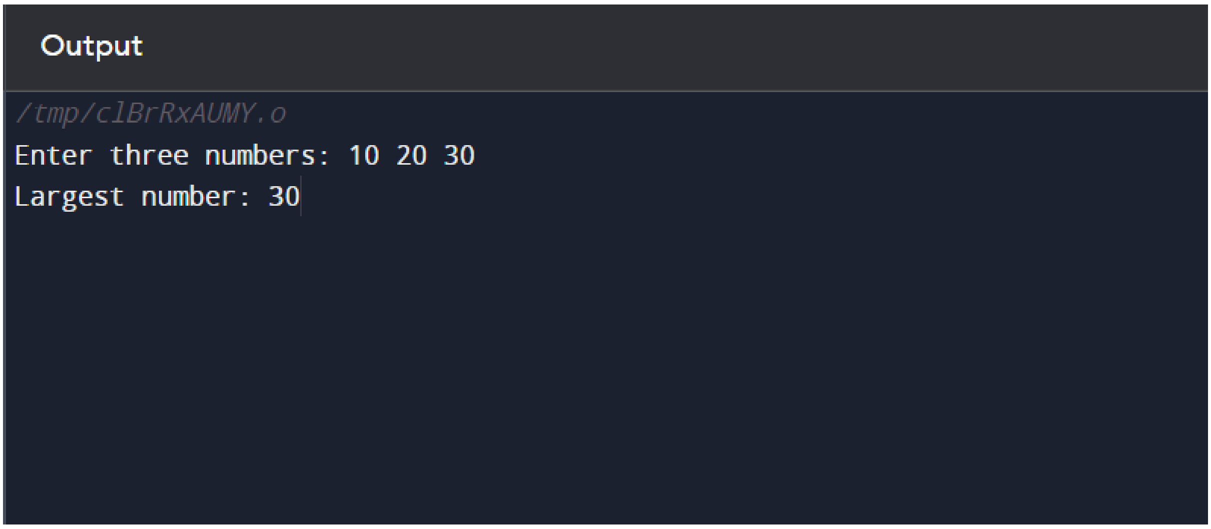
```
Output
/tmp/clBrRxAUMY.o
1
1
1
0
```

. Write C ++ Program To Check The Largest Number Among Three Given Integer.

```
#include <iostream>
using namespace std;

int main() {
   float n1, n2, n3;

   cout << "Enter three numbers: ";
```

```
    cin >> n1 >> n2 >> n3;

    if(n1 >= n2 && n1 >= n3)
        cout << "Largest number: " << n1;

    if(n2 >= n1 && n2 >= n3)
        cout << "Largest number: " << n2;

    if(n3 >= n1 && n3 >= n2)
        cout << "Largest number: " << n3;

    return 0;
}
```

OUTPUT :

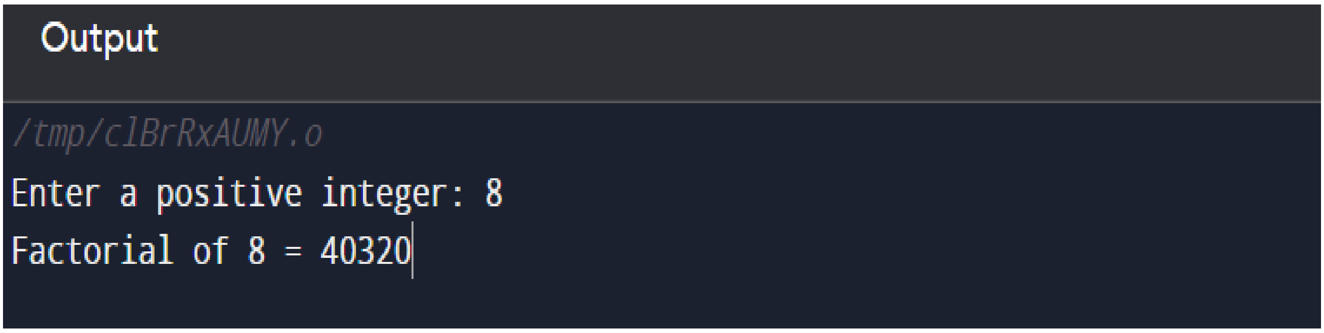


```
/tmp/clBrRxAUMY.o
Enter three numbers: 10 20 30
Largest number: 30
```

. Write A Program In C++ TO Find Factorial Of A Number.

```
#include <iostream>
using namespace std;

int main() {
    int n;
    long double factorial = 1.0;

    cout << "Enter a positive integer: ";
    cin >> n;

    if (n < 0)
        cout << "Error! Factorial of a negative number doesn't exist.";
    else {
        for(int i = 1; i <= n; ++i) {
            factorial *= i;
        }
        cout << "Factorial of " << n << " = " << factorial;
    }

    return 0;
}
```

OUTPUT :

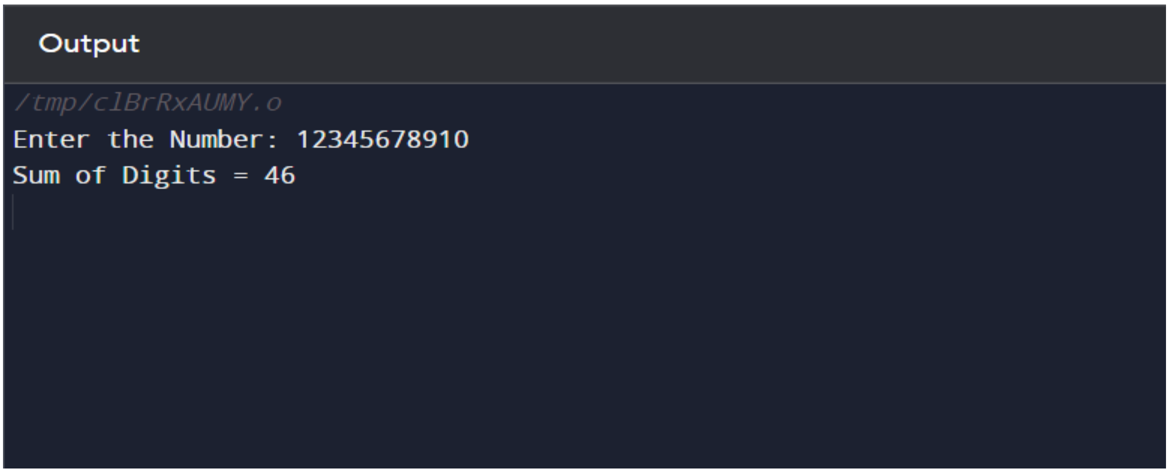


```
/tmp/clBrRxAUMY.o
Enter a positive integer: 8
Factorial of 8 = 40320
```

. Write a program in C++ to find the sum of digits of a given number.

```
#include<iostream>
using namespace std;
int main()
{
   int num, rem, sum=0;
   cout<<"Enter the Number: ";
   cin>>num;
   while(num>0)
   {
      rem = num%10;
      sum = sum+rem;
      num = num/10;
   }
   cout<<"\nSum of Digits = "<<sum;
   cout<<endl;
   return 0;
}
```

OUTPUT :

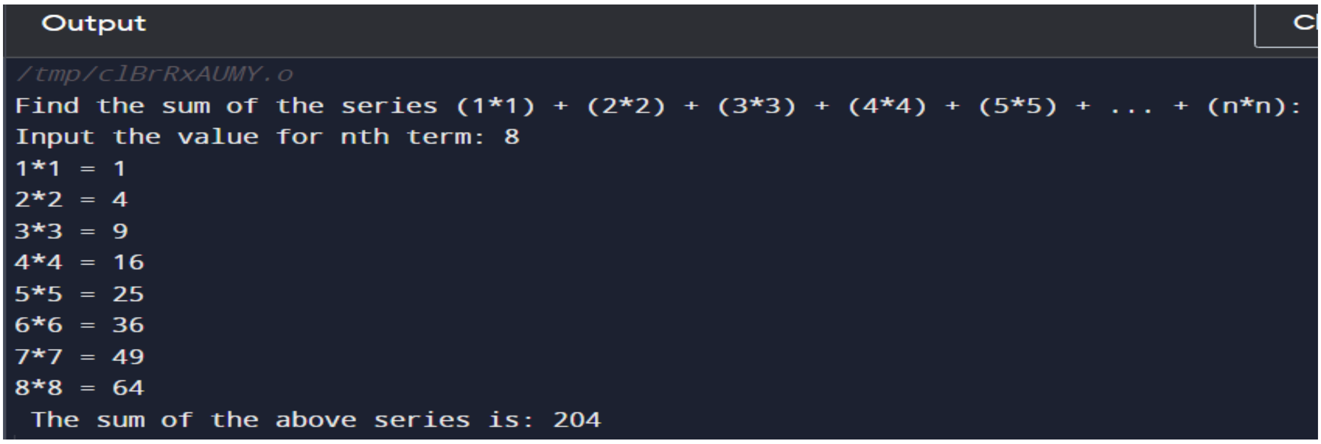


```
/tmp/clBrRxAUMY.o
Enter the Number: 12345678910
Sum of Digits = 46
```

. Write a program in C++ to calculate the sum of the series (1*1) + (2*2) + (3*3) + (4*4) + (5*5) + ... + (n*n).

```
#include <iostream>
using namespace std;

int main()
{
   int i, n, sum = 0;
   cout << "\n\n Find the sum of the series (1*1) + (2*2) + (3*3) + (4*4) + (5*5) 
+ ... + (n*n):\n";
   cout << " Input the value for nth term: ";
   cin >> n;

   for (i = 1; i <= n; i++)
     {
      sum += i * i;
      cout << i << "*" << i << " = " << i * i << endl;
   }
   cout << " The sum of the above series is: " << sum << endl;
}
```

OUTPUT :

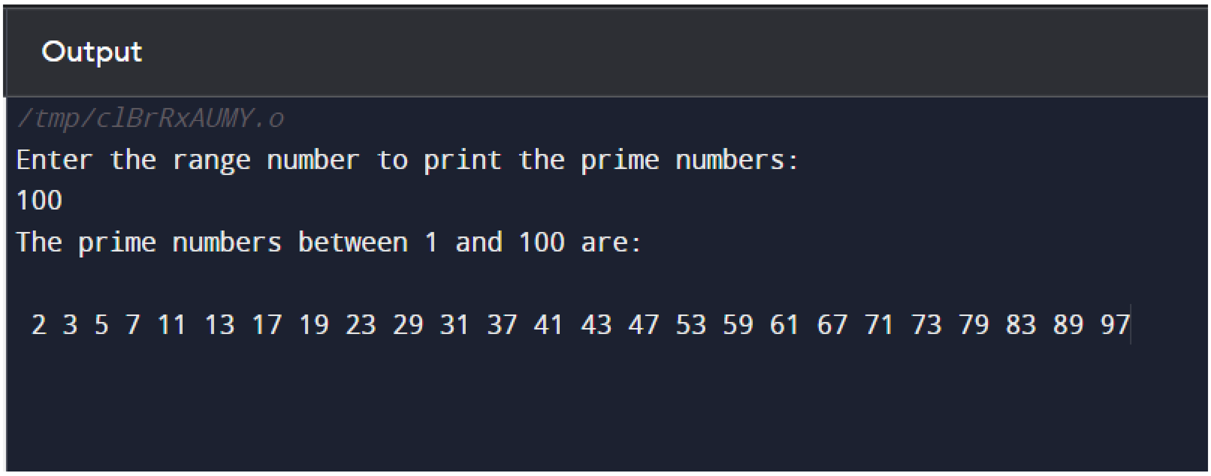


```
/tmp/clBrRxAUMY.o
Find the sum of the series (1*1) + (2*2) + (3*3) + (4*4) + (5*5) + ... + (n*n):
Input the value for nth term: 8
1*1 = 1
2*2 = 4
3*3 = 9
4*4 = 16
5*5 = 25
6*6 = 36
7*7 = 49
8*8 = 64
 The sum of the above series is: 204
```

. C++ Program to Print Prime Numbers from 1 to N using For loop .

```
#include <iostream>
using namespace std;

int main() {
    int x, i, j, f;

    cout << "Enter the range number to print the prime numbers:\n";
    cin >> x;

    cout << "\n\nThe prime numbers between 1 and " << x << " are:\n\n";

    for (i = 1; i <= x; i++) {


        if (i == 1 || i == 0) {
            continue;
        }

        f = 1;

        for (j = 2; j <= i / 2; ++j) {
            if (i % j == 0) {
                f = 0;
                break;
            }
        }
if (f == 1) {
            cout << " " << i;
        }
    }
    return 0;
}
```

OUTPUT :

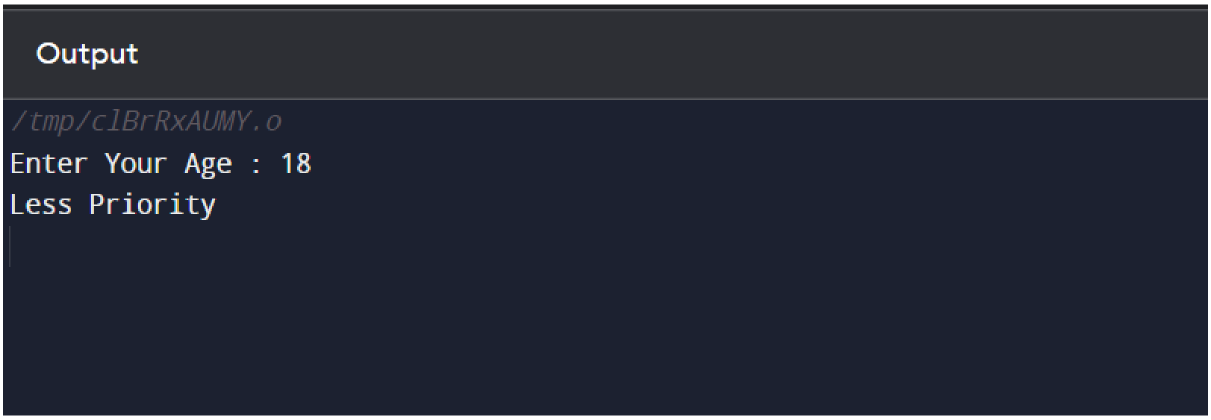


```
/tmp/clBrRxAUMY.o
Enter the range number to print the prime numbers:
100
The prime numbers between 1 and 100 are:

 2 3 5 7 11 13 17 19 23 29 31 37 41 43 47 53 59 61 67 71 73 79 83 89 97
```

. Write A Program For The Following Case : Give Age Of A Person As Input . Then Print Priority Of Covid Vaccinattion According To Their Age.Age<18 Not Eligible , Age>60 Eligible And Highest Priority , 18 < Age <60 Eligible With Less Priority.

```
#include <iostream>
using namespace std;

int main() {
 int age;

 cout << "Enter Your Age : ";
 cin >> age;

 if (age < 18) {
   cout << "Not Eligible For Vaccination"<< endl;
 }
 else if(age>=18 && age<=60){
     cout << "Less Priority"<< endl;
 }
 else{
     cout << "Highest Priority For Vaccination"<< endl;
 }

 return 0;
}
```

OUTPUT :

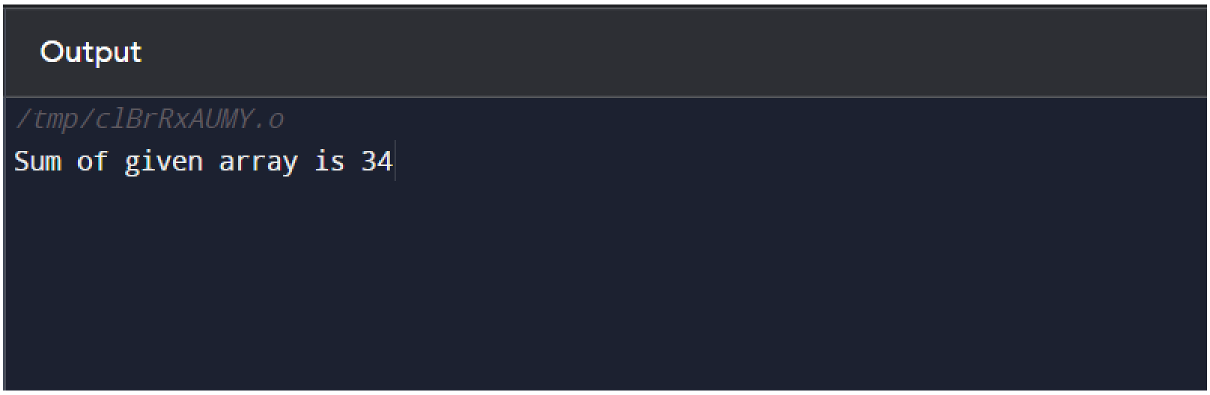


```
/tmp/clBrRxAUMY.o
Enter Your Age : 18
Less Priority
```

. Find Sum Of Array Elements .

```
#include <bits/stdc++.h>
using namespace std;
int sum(int arr[], int n)
{
    int sum = 0;

    for (int i = 0; i < n; i++)
    sum += arr[i];

    return sum;
}
int main()
{
    int arr[] = {12, 3, 4, 15};
    int n = sizeof(arr) / sizeof(arr[0]);
    cout << "Sum of given array is " << sum(arr, n);
    return 0;
}
```

OUTPUT :



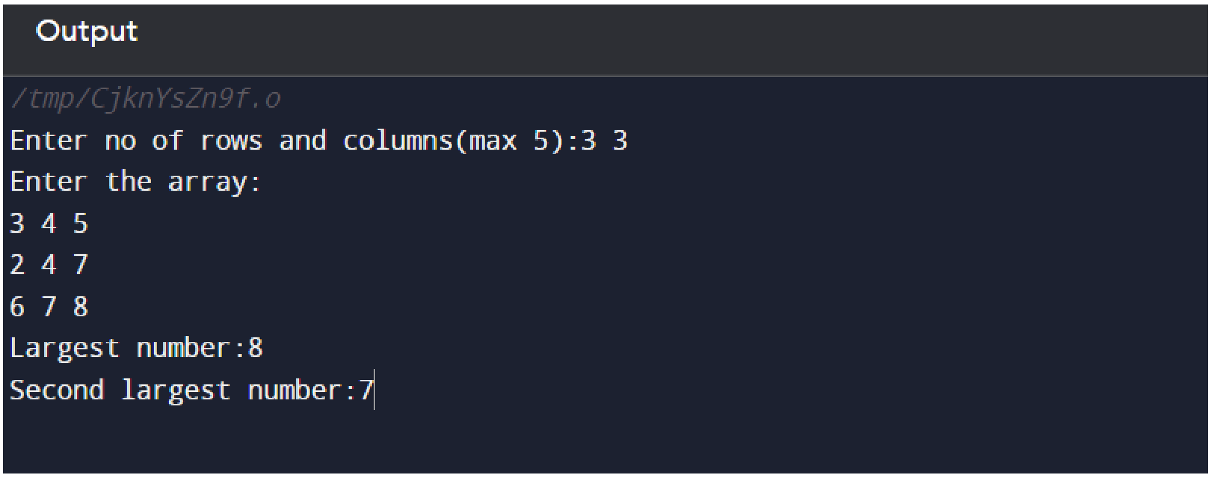
```
/tmp/clBrRxAUMY.o
Sum of given array is 34
```

. Find The Largest And Second Largest Array Of 2D Array.

```
#include<iostream>

using namespace std;

int main()
{
    int a[5][5],big1=1,big2=0,n,m,i,j;
    cout<<"Enter no of rows and columns(max 5):";
    cin>>m>>n;
    cout<<"Enter the array:\n";

    for(i=0;i<m;i++)
        for(j=0;j<n;++j)
            cin>>a[i][j];

    for(i=0;i<m;++i)
        for(j=0;j<n;++j)
        {
            if(a[i][j]>big1)
                big1=a[i][j];
        }

    for(i=0;i<m;++i)
        for(j=0;j<n;++j)
        {
            if(a[i][j]>big2&&a[i][j]<big1)
                big2=a[i][j];
```

```
      }

   cout<<"\nLargest number:"<<big1;
   cout<<"\nSecond largest number:"<<big2;



   return 0;
}
```

OUTPUT :

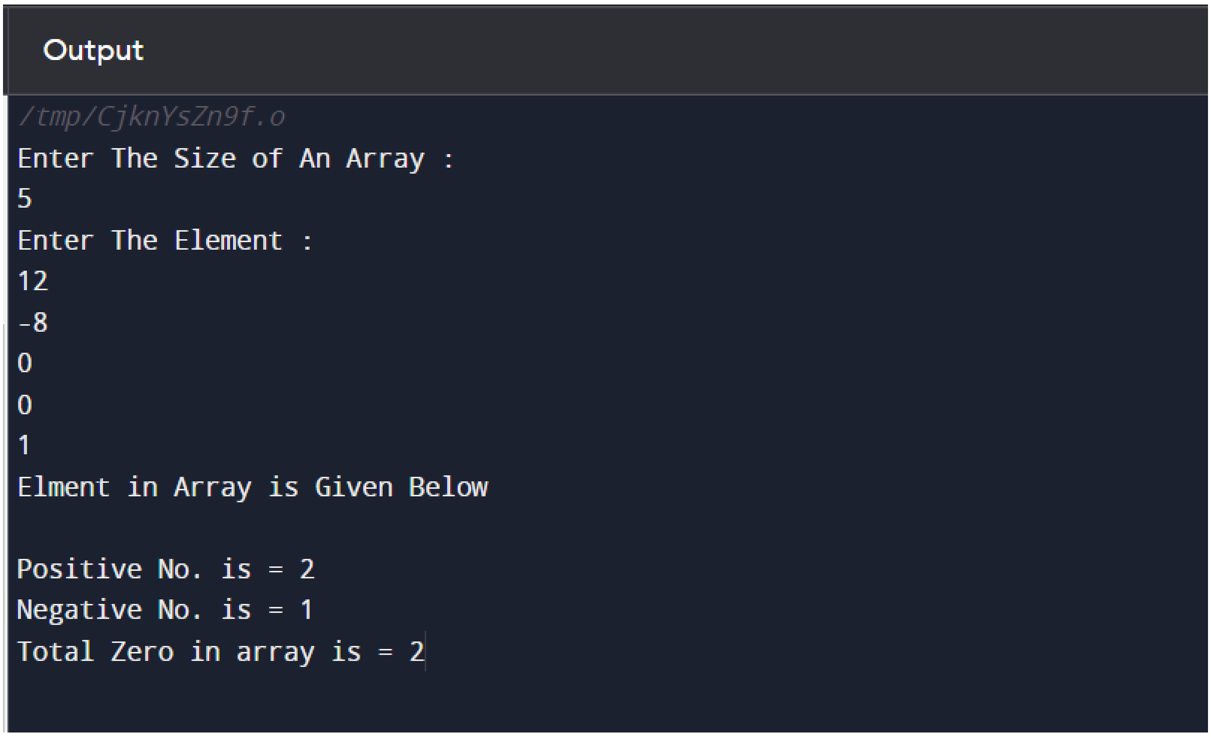


```
Output
/tmp/CjknYsZn9f.o
Enter no of rows and columns(max 5):3 3
Enter the array:
3 4 5
2 4 7
6 7 8
Largest number:8
Second largest number:7
```

. Count Number Of Positive, Negative, And Zeroes Number In An Array.

```
#include<iostream>
using namespace std;
int main()
{
  int a[100],i,n,zero=0,pos=0,neg=0;
cout<<"Enter The Size of An Array :\n";
  cin>>n;

  cout<<"Enter The Element :\n";
  for(i=0;i<n;i++)
  {
   cin>>a[i];
  }

  cout<<"Elment in Array is Given Below\n";
  for(i=0;i<n;i++)
  {
   if(a[i]>0)
   pos++;
   else if(a[i]<0)
   neg++;
   else
   zero++;
  }
  cout<<"\nPositive No. is = "<<pos;
  cout<<"\nNegative No. is = "<<neg;
  cout<<"\nTotal Zero in array is = "<<zero;
return 0;

}
```

OUTPUT :



Output

```
/tmp/CjknYsZn9f.o
Enter The Size of An Array :
5
Enter The Element :
12
-8
0
0
1
Elment in Array is Given Below

Positive No. is = 2
Negative No. is = 1
Total Zero in array is = 2
```

. What are sparse, upper triangular and lower matrix .

```
#include <iostream>
using namespace std;

int main()
{
        int n;
        cin >> n;
        int flag = 0;
        int mat[n][n];
        int i, j;
        for(i = 0; i < n; i++)
        {
        for(j = 0; j < n; j++)
        cin >>mat[i][j];
```

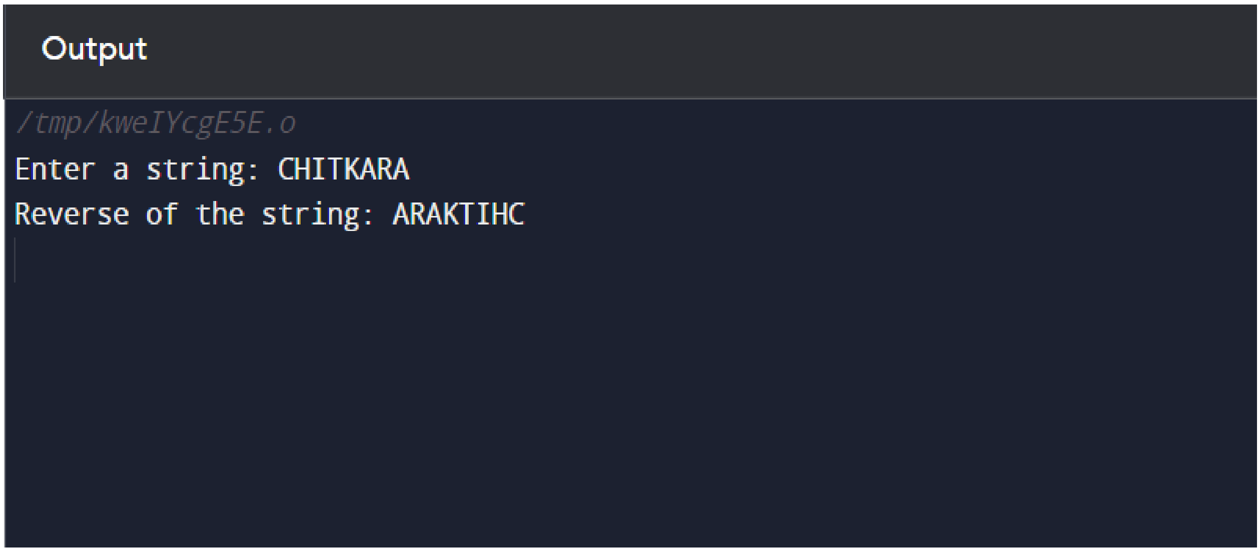
```
    }

for (i = 1; i < n; i++)
    for (j = 0; j < i; j++)
    if (mat[i][j] != 0)
    flag = 0;
    else
    flag = 1;

    if (flag == 1)
    cout << “Upper Triangular Matrix”;
    else
    cout << “Not an Upper Triangular Matrix”;
    return 0;
}
```

. Reverse A String Using C++

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

void reverseString(char* str)
{
    int l, i;
    char *begin_ptr, *end_ptr, ch;


    l = strlen(str);
    begin_ptr = str;
    end_ptr = str;



    for (i = 0; i < l - 1; i++)
    end_ptr++;



    for (i = 0; i < l / 2; i++) {



    ch = *end_ptr;
    *end_ptr = *begin_ptr;
    *begin_ptr = ch;

    begin_ptr++;
    end_ptr--;
    }
}



int main()
{



    char str[100] = "CHITKARA";
    printf("Enter a string: %s\n", str);
```

```
        reverseString(str);



        printf("Reverse of the string: %s\n", str);


        return 0;
}
```

OUTPUT :



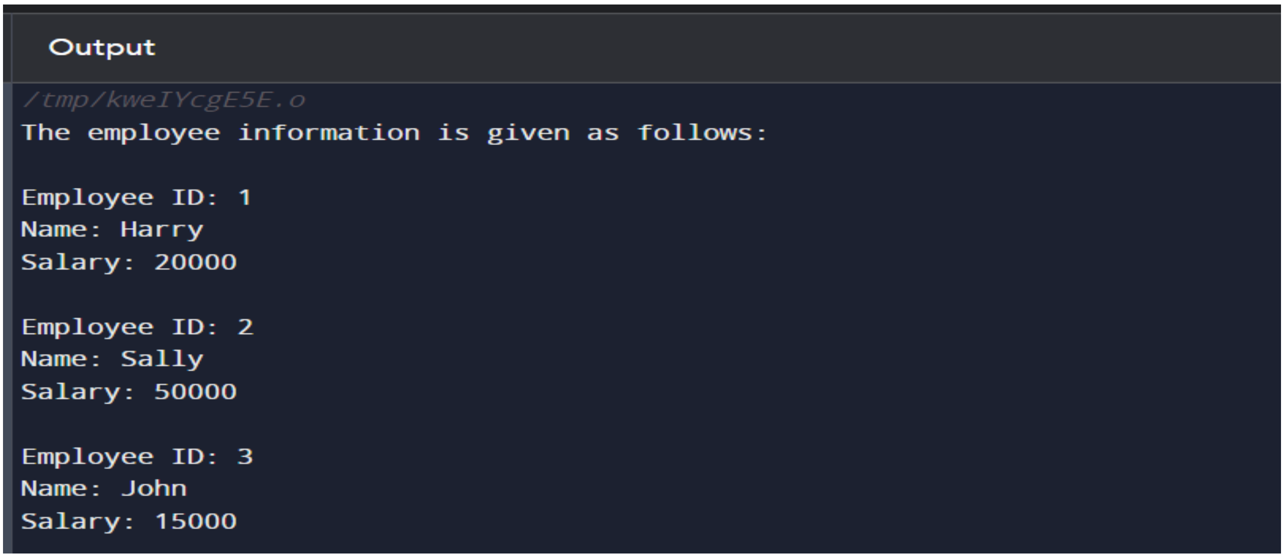
Output

```
/tmp/kweIYcgE5E.o
Enter a string: CHITKARA
Reverse of the string: ARAKTIHC
```

. WAP for creating employee data (uid, name, salary) using constructor.

```
#include <iostream>
using namespace std;
struct employee
{
int ID;
char name[50];
int salary;
};
int main()
{
struct employee emp[3] = {{1, "Harry", 20000,}, {2, "Sally", 50000,}, {3, "John",
15000,}};
cout << "The employee information is given as follows:" << endl;
cout << endl;
for (int i = 0; i < 3; i++)
{
cout << "Employee ID: " << emp[i].ID << endl;
cout << "Name: " << emp[i].name << endl;
cout << "Salary: " << emp[i].salary << endl;
cout << endl;
}
return 0;
}
```

OUTPUT :

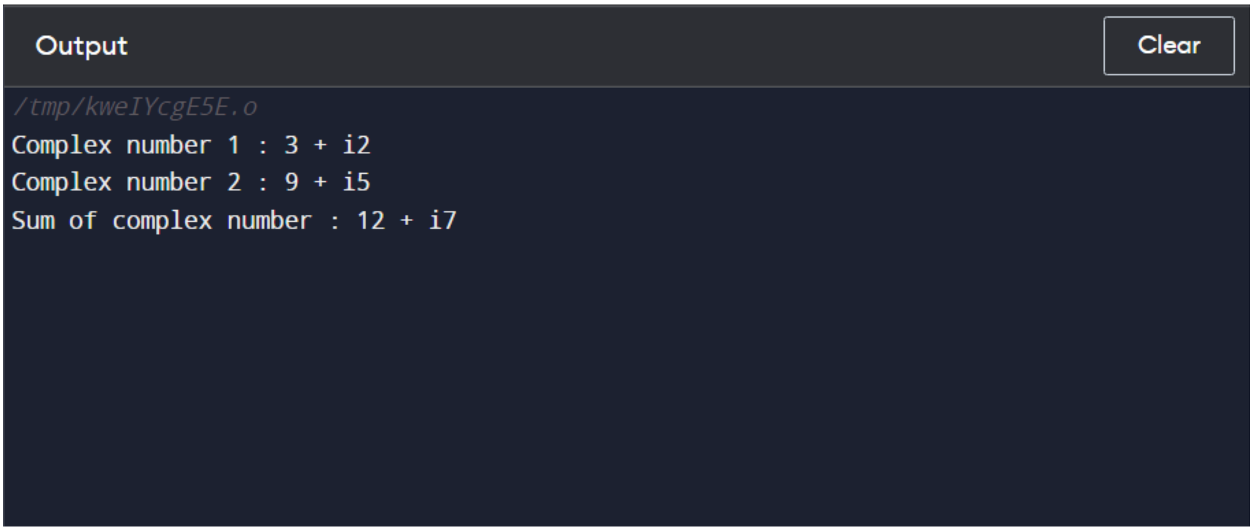


```
/tmp/kweIYcgE5E.o
The employee information is given as follows:

Employee ID: 1
Name: Harry
Salary: 20000

Employee ID: 2
Name: Sally
Salary: 50000

Employee ID: 3
Name: John
Salary: 15000
```

. WAP to add two complex numbers using constructor.

```
#include <bits/stdc++.h>
using namespace std;
class Complex
{
public:
int real, imaginary;
Complex(int tempReal = 0, int tempImaginary = 0)
{
real = tempReal;
imaginary = tempImaginary;
}
Complex addComp(Complex C1, Complex C2)
{
Complex temp;
temp.real = C1.real + C2.real;
temp.imaginary = C1.imaginary + C2.imaginary;
return temp;
}
};
int main()
{
Complex C1(3, 2);
cout << "Complex number 1 : " << C1.real
<< " + i" << C1.imaginary << endl;
Complex C2(9, 5);
cout << "Complex number 2 : " << C2.real
<< " + i" << C2.imaginary << endl;
Complex C3;
C3 = C3.addComp(C1, C2);
cout << "Sum of complex number : "
<< C3.real << " + i"
<< C3.imaginary;
}
```

OUTPUT :



```
/tmp/kweIYcgE5E.o
Complex number 1 : 3 + i2
Complex number 2 : 9 + i5
Sum of complex number : 12 + i7
```

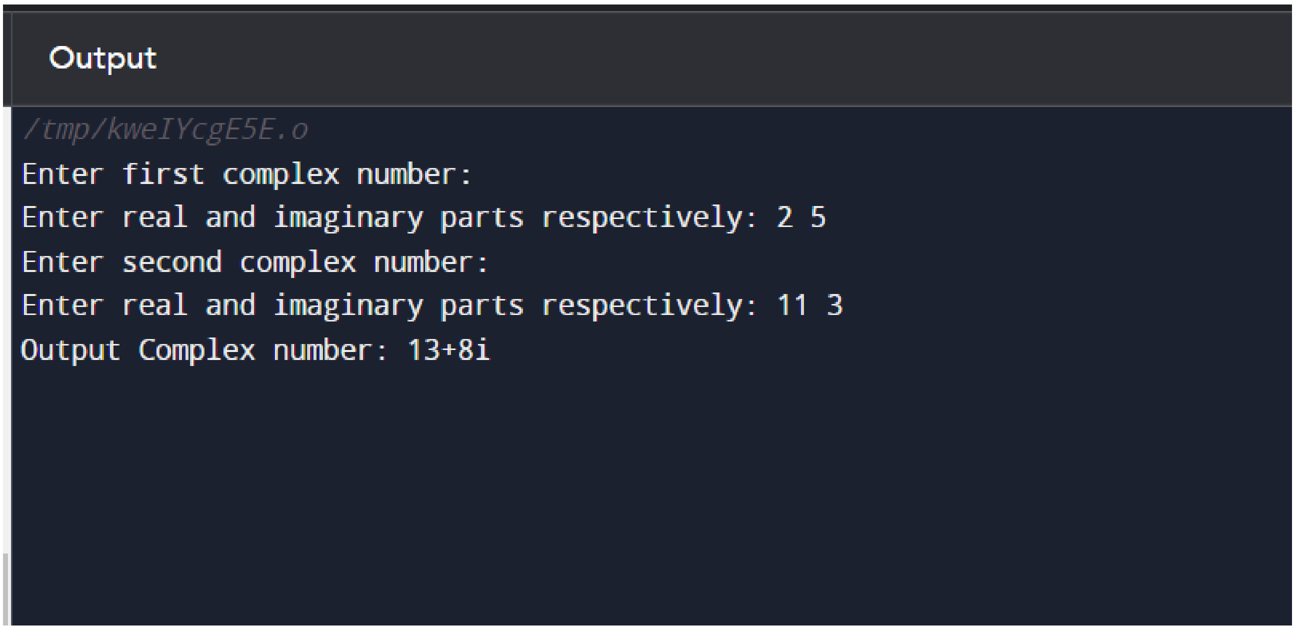
. WAP to overload binary operator in a class.

```
#include <iostream>
using namespace std;
class Complex
{
private:
float real;
float imag;
public:
Complex() : real(0), imag(0) {}
void input()
```

```
{
cout << "Enter real and imaginary parts respectively: ";
cin >> real;
cin >> imag;
}
Complex operator+(const Complex &obj)
{
Complex temp;
temp.real = real + obj.real;
temp.imag = imag + obj.imag;
return temp;
}
void Output()
{
if (imag < 0)
cout << "Output Complex number: " << real << imag << "i";
else
cout << "Output Complex number: " << real << "+" << imag << "i";
}
};
int main()
{
Complex complex1, complex2, result;
cout << "Enter first complex number:\n";
complex1.input();
cout << "Enter second complex number:\n";
complex2.input();
result = complex1 + complex2;
result.Output();
```

```
return 0;}
```

OUTPUT :



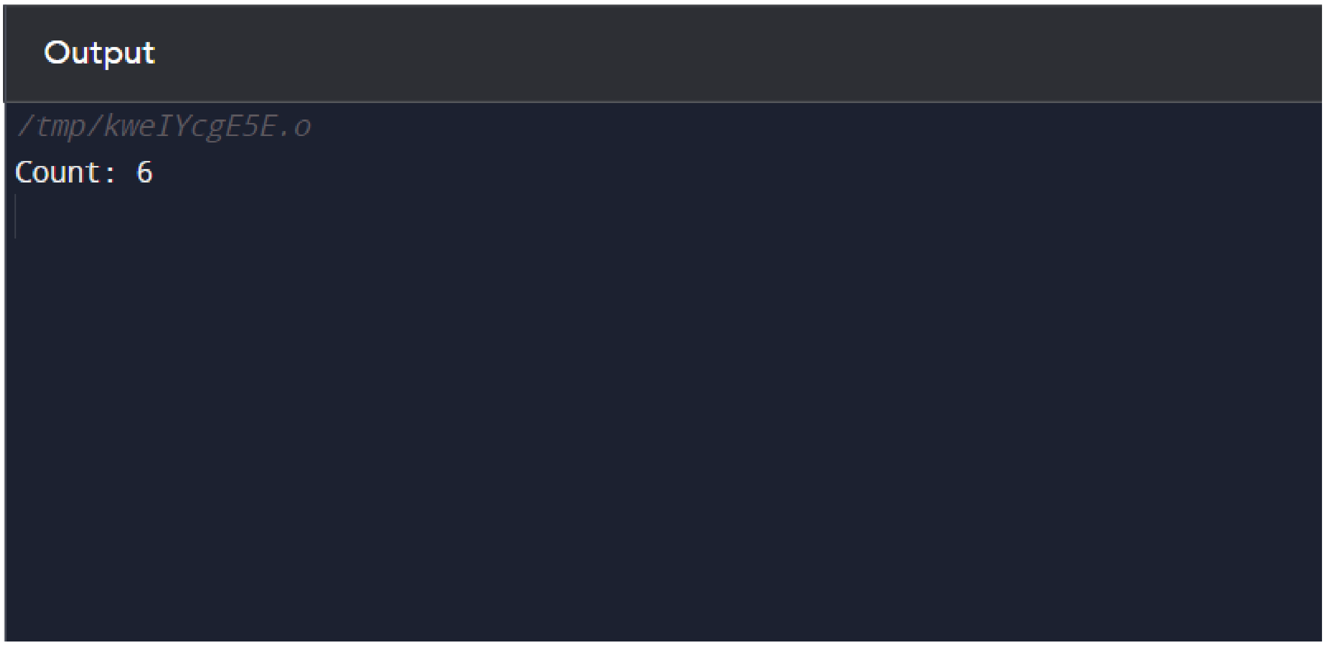
```
/tmp/kweIYcgE5E.o
Enter first complex number:
Enter real and imaginary parts respectively: 2 5
Enter second complex number:
Enter real and imaginary parts respectively: 11 3
Output Complex number: 13+8i
```

. WAP to overload unary operator in a class.

```
#include <iostream>
using namespace std;
class Count
{
private:
int value;
public:

Count() : value(5) {}

void operator++()
{
```

```
++value;
}
void display()
{
cout << "Count: " << value << endl;
}
};
int main()
{
Count count1;

++count1;
count1.display();
return 0;
}
```

OUTPUT :



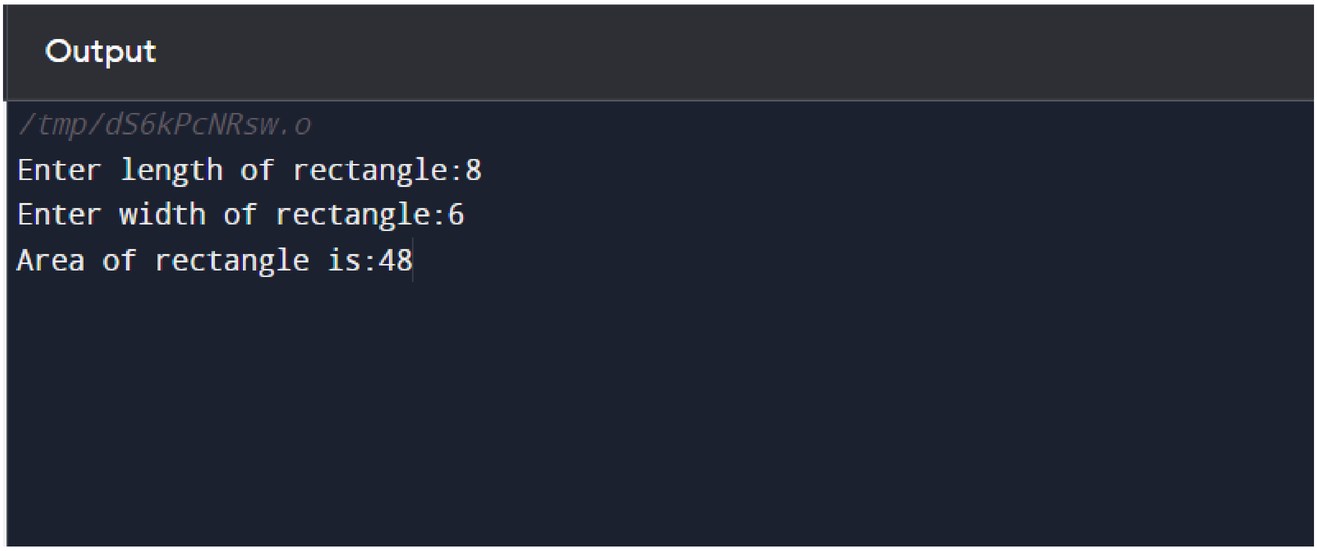
```
/tmp/kweIYcgE5E.o
Count: 6
```

. WAP TO FETCH THE AREA OF RECTANGLE USING CLASSES.

```
#include<iostream>
using namespace std;

class Test {
public:
    int length, width, area;

    void input() {
        cout << "Enter length of rectangle:";
        cin >> length;
        cout << "Enter width of rectangle:";
        cin>>width;
    }

    void findArea() {
        area = length * width;
    }

    void display() {
        cout << "Area of rectangle is:" << area;
    }
};

int main() {

    Test obj;

```

```
   obj.input();
   obj.findArea();
   obj.display();

   return 0;
}
```

OUTPUT :

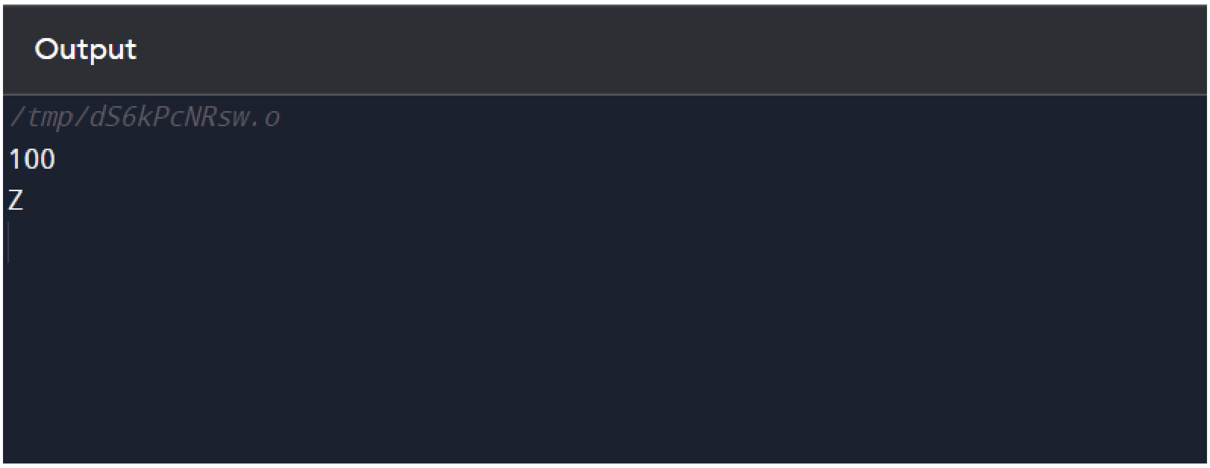


```
/tmp/dS6kPcNRsw.o
Enter length of rectangle:8
Enter width of rectangle:6
Area of rectangle is:48
```

. Friend Function Example.

```
#include <iostream>
using namespace std;
class XYZ
{
private:
   int num=100;
   char ch='Z';
public:
   friend void disp(XYZ obj);
};
//Global Function

void disp(XYZ obj)
{
   cout<<obj.num<<endl;
   cout<<obj.ch<<endl;
}



int main()
 {
   XYZ obj;
   disp(obj);
   return 0;
}
```

OUTPUT :

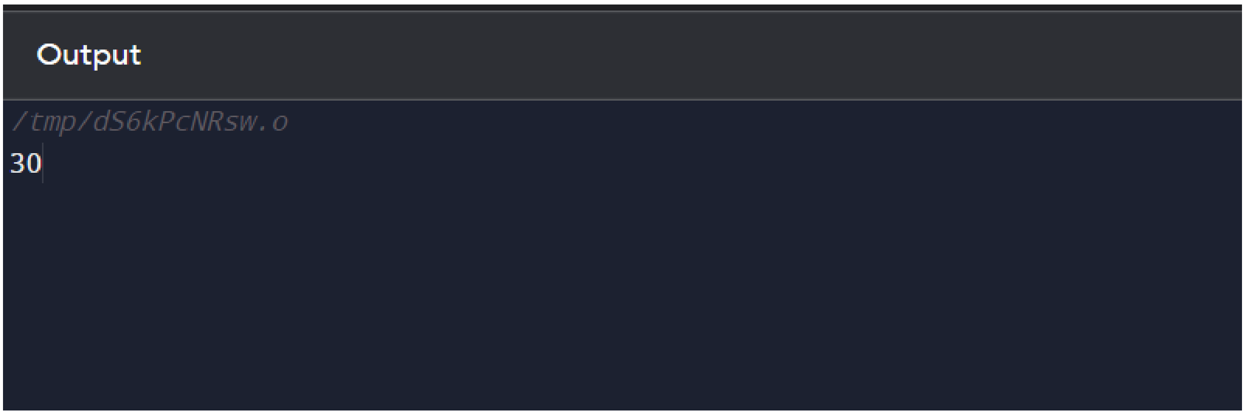


```
/tmp/dS6kPcNRsw.o
100
Z
```

. Function With Const.

```
 #include <iostream>
using namespace std;
class Test {
    int value;
public:
    void putValue1()
    {
        value = 30;
    }
    int getValue() {return value; }
};
int main()
{
    Test t;
    t.putValue1();
    cout<<t.getValue();
    return 0;
}
```

OUTPUT :

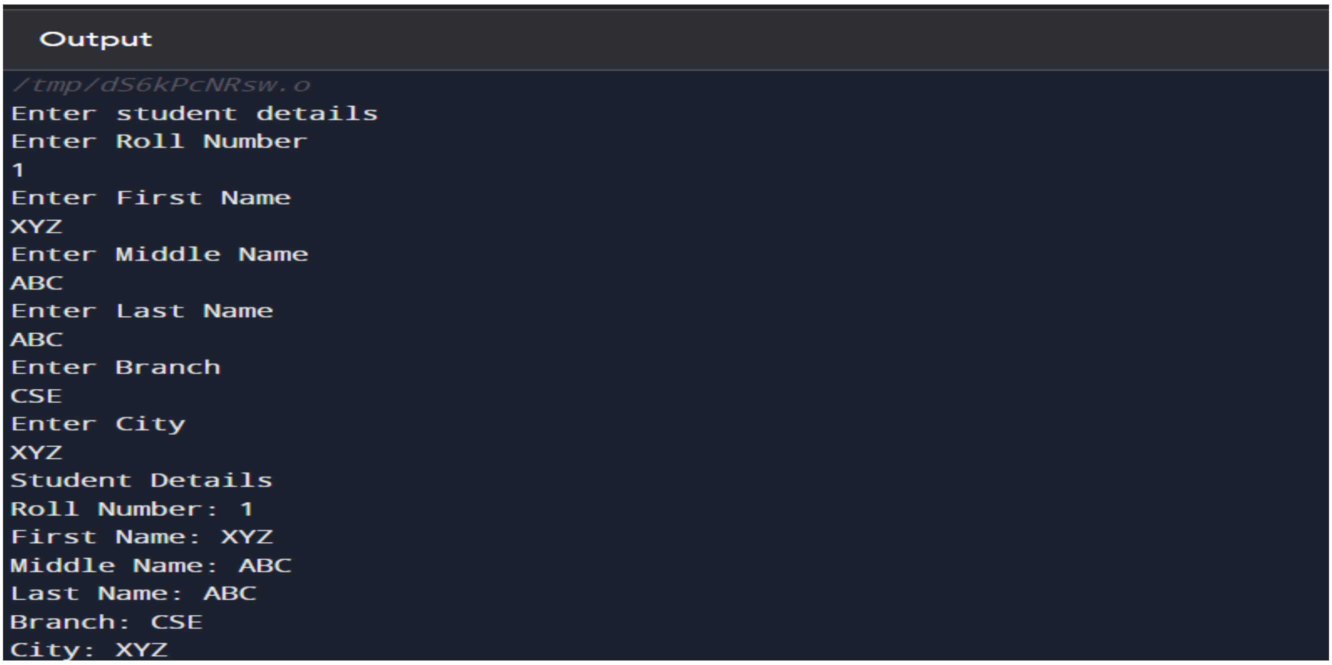


```
/tmp/dS6kPcNRsw.o
30
```

. C++ Program to Read and Display Student Details using Class.

```
#include <iostream>
using namespace std;
class student
{
private:
    int rollno;
    char fname[10],mname[10],lname[10];
    char branch[15];
    char city[15];
public:
    void get_data()
    {
        cout<<"Enter Roll Number"<<endl;
        cin>>rollno;
        cout<<"Enter First Name"<<endl;
        cin>>fname;
        cout<<"Enter Middle Name"<<endl;
        cin>>mname;
        cout<<"Enter Last Name"<<endl;
        cin>>lname;
        cout<<"Enter Branch"<<endl;
        cin>>branch;
        cout<<"Enter City"<<endl;
        cin>>city;
    }
    void display()
    {
        cout<<"Roll Number: "<<rollno<<endl;
        cout<<"First Name: "<<fname<<endl;
        cout<<"Middle Name: "<<mname<<endl;
        cout<<"Last Name: "<<lname<<endl;
        cout<<"Branch: "<<branch<<endl;
        cout<<"City: "<<city<<endl;
    }
};
int main()
{
    student s1;
    cout << "Enter student details" << endl;
    s1.get_data();
    cout << "Student Details " << endl;
    s1.display();
    return 0;
}
```

OUTPUT :

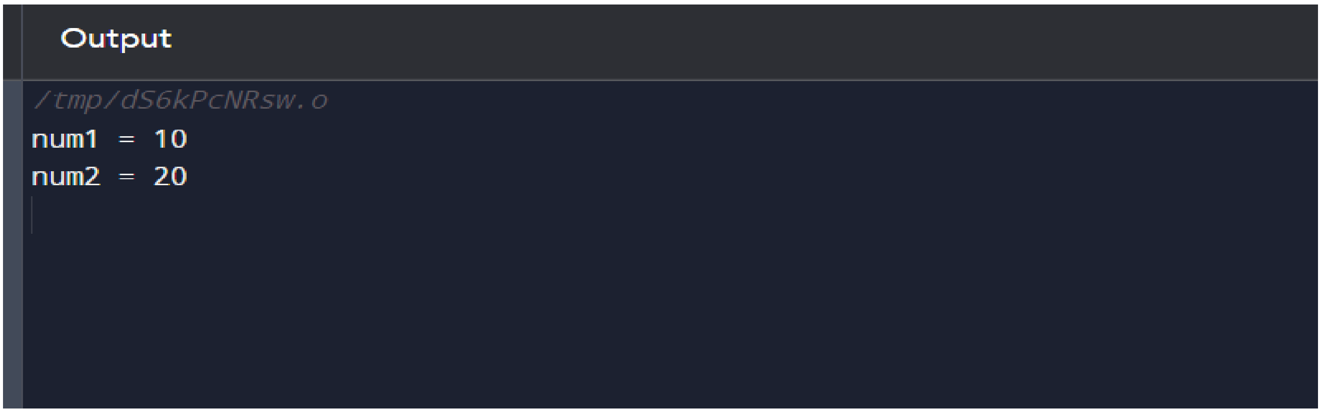


```
/tmp/dS6kPcNRsw.o
Enter student details
Enter Roll Number
1
Enter First Name
XYZ
Enter Middle Name
ABC
Enter Last Name
ABC
Enter Branch
CSE
Enter City
XYZ
Student Details
Roll Number: 1
First Name: XYZ
Middle Name: ABC
Last Name: ABC
Branch: CSE
City: XYZ
```

. Default Constructor Example .

```
#include <iostream>
using namespace std;
class Demo {
   private:
   int num1, num2 ;
   public:
   Demo() {
      num1 = 10;
      num2 = 20;
   }
   void display() {
      cout<<"num1 = "<< num1 <<endl;
      cout<<"num2 = "<< num2 <<endl;
   }   };
int main() {
   Demo obj;
   obj.display();
   return 0;
}
```

OUTPUT :

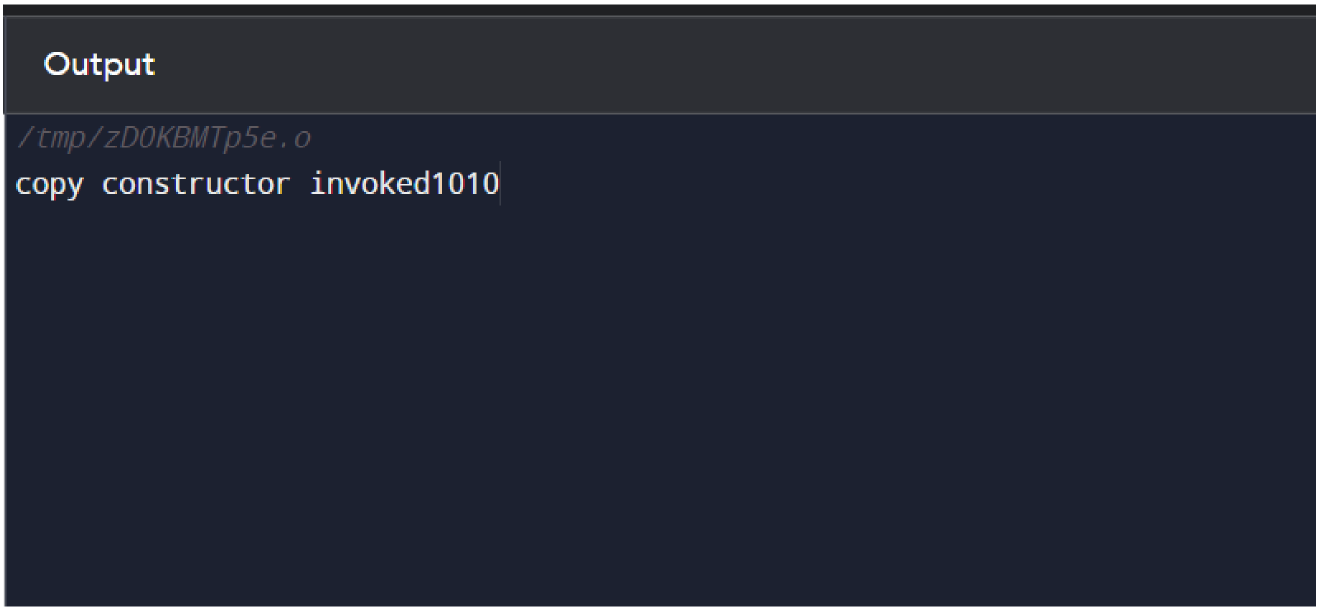


```
/tmp/dS6kPcNRsw.o
num1 = 10
num2 = 20
```

. Copy Constructor Example.

```
#include <iostream>
using namespace std;

class counter
{
int c;
public:
counter(int a)
{
c=a;
}
counter(counter &ob)
{
cout<<“copy constructor invoked”;
c=ob.c;
}

void show()
{
cout<<c;
}
};
int main()
{
counter C1(10);
counter C2(C1);
C1.show();
C2.show();
return 0;
}
```

OUTPUT :

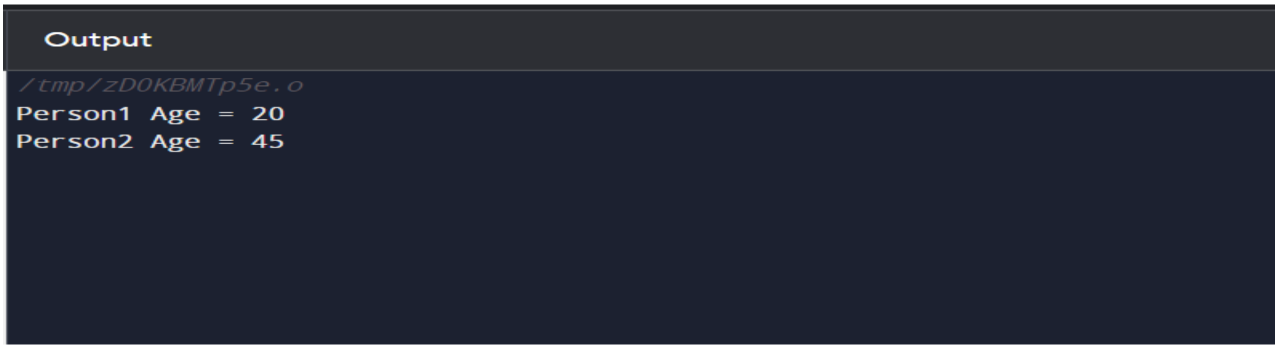


```
/tmp/zDOKBMTp5e.o
copy constructor invoked1010
```

. Overloading Constructors Example.

```
 #include<iostream>
using namespace std;
class Person{
private:
int age;
public:

Person()
{
age=20;
}

Person(int a)
{

age=a;
}
int getAge()
{
return age;
}
};
int main()
{
Person person1, person2(45);

cout<< "Person1 Age = "<<person1.getAge()<<endl;
cout<< "Person2 Age = "<<person2.getAge()<<endl;
return 0;
}
```

OUTPUT :

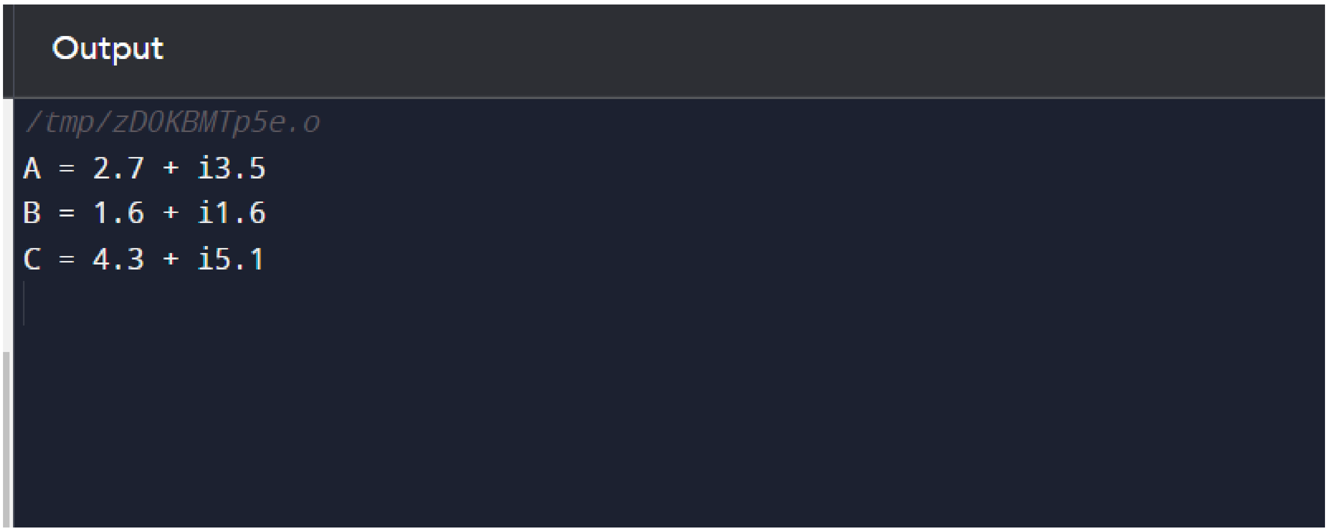


```
/tmp/zD0KBMTp5e.o
Person1 Age = 20
Person2 Age = 45
```

. **WAP to implement the use of constructor overloading in a class.**

```
#include<iostream>
using namespace std;
class complex{
float x,y;
public:
complex(){ }
complex(float a){x=y=a;}
complex(float real, float imag)
{x=real; y=imag;}
friend complex sum(complex c1, complex c2)
{
complex c3;
c3.x = c1.x + c2.x;
c3.y = c1.y + c2.y;
return(c3);
}
friend void show(complex c)
{cout<<c.x<<" + i"<<c.y<<"\n";}
};

int main()
{
complex A(2.7, 3.5);
complex B(1.6);
complex C;
C=sum(A,B);
cout<<"A = ";show(A);
cout<<"B = ";show(B);
cout<<"C = ";show(C);
return 0;
}
```

OUTPUT :



```
/tmp/zDOKBMTp5e.o
A = 2.7 + i3.5
B = 1.6 + i1.6
C = 4.3 + i5.1
```