

LAB 1**1. WAP to print welcome to C++**

```
#include<iostream>

using namespace std;

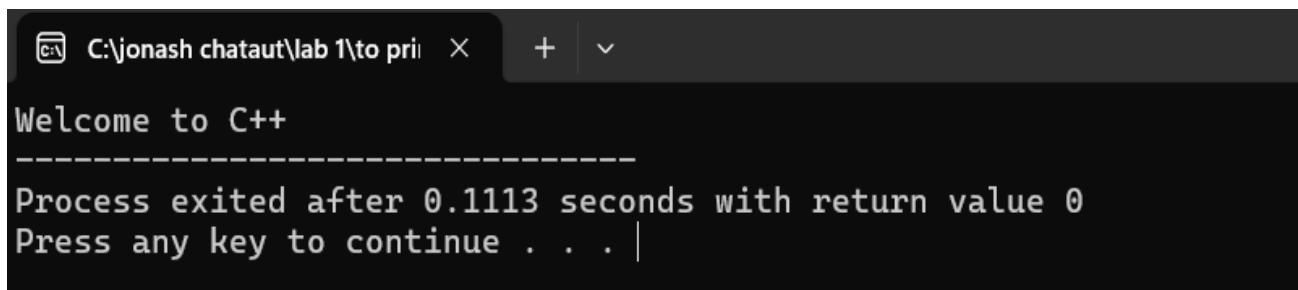
int main()

{

    cout<<"Welcome to C++";

    return 0;

}
```

Output:

A screenshot of a terminal window titled "C:\jonash chataut\lab 1\to pri". The window displays the following text:
Welcome to C++

Process exited after 0.1113 seconds with return value 0
Press any key to continue . . . |

2. WAP to find size of different datatypes in C++

```
#include<iostream>

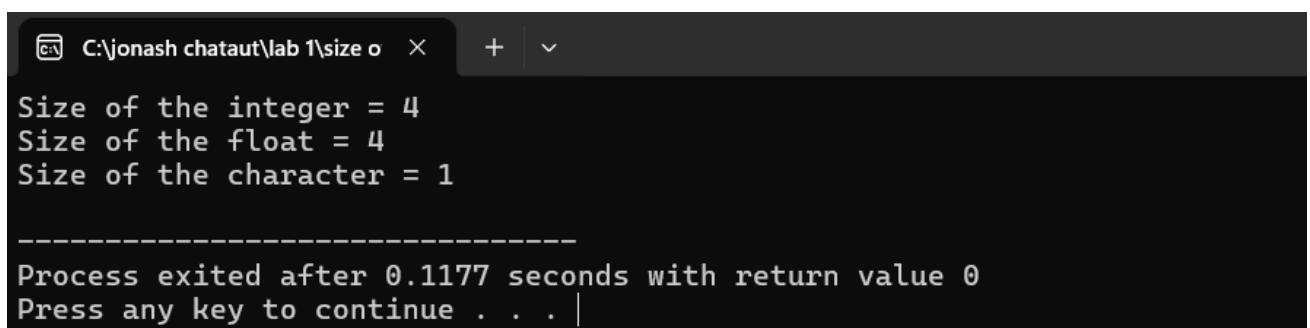
using namespace std;

int main() {

    cout<<"Size of the integer = "<<sizeof(int)<<endl;
    cout<<"Size of the float = "<<sizeof(float)<<endl;
    cout<<"Size of the character = "<<sizeof(char)<<endl;

    return 0;

}
```

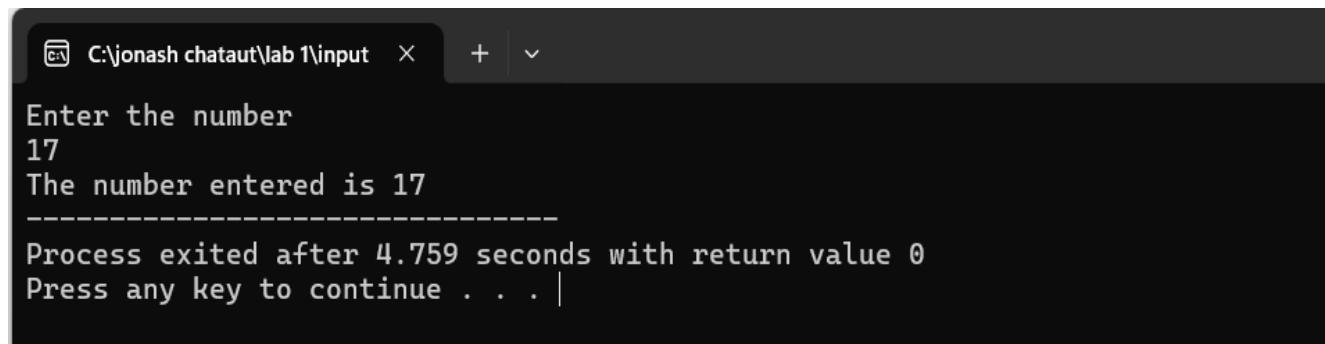
Output:

A screenshot of a terminal window titled "C:\jonash chataut\lab 1\size o". The window displays the following text:
Size of the integer = 4
Size of the float = 4
Size of the character = 1

Process exited after 0.1177 seconds with return value 0
Press any key to continue . . . |

LAB 1**3. WAP to input number and display it.**

```
#include<iostream>
using namespace std;
int main()
{
    int n;
    cout<<"Enter the number"<<endl;
    cin>>n;;
    cout<<"The number entered is "<<n;
    return 0;
}
```

Output:

```
C:\jonash chataut\lab 1\input  x  +  ▾
Enter the number
17
The number entered is 17
-----
Process exited after 4.759 seconds with return value 0
Press any key to continue . . . |
```

4. WAP to find the sum of two numbers input by user.

```
#include<iostream>
using namespace std;
int main()
{
    int a,b;
    cout<<"Enter two numbers ";
    cin>>a>>b;
    int sum;
    sum=a+b;
    cout<<"The sum of given two number is "<<sum;
    return 0;
}
```

LAB 1**Output:**

```
C:\jonash chataut\lab 1\sum c  X  +  ▾  
Enter two numbers 4  
5  
The sum of given two number is 9  
-----  
Process exited after 3.832 seconds with return value 0  
Press any key to continue . . . |
```

5. WAP to find SI

```
#include<iostream>  
using namespace std;  
int main()  
{  
    int r,t,p;  
    float SI;  
    cout<<"Enter the principal value, time and rate" << endl;  
    cin>>p>>t>>r;  
    SI=(p*t*r)/100;  
    cout<<"The simple interest is " << SI << endl;  
    return 0;  
}
```

Output:

```
C:\jonash chataut\lab 1\WAP 1  X  +  ▾  
Enter the principal value, time and rate  
100  
45  
2  
The simple interest is 90  
-----  
Process exited after 11.4 seconds with return value 0  
Press any key to continue . . . |
```

LAB 2**1. WAP to find the prime numbers upto n values entered by user.**

```
#include<iostream>
using namespace std;
int main()
{
    int n,count=0;
    cout<<"Enter the value upto which you want the prime numbers:"<<endl;
    cin>>n;
    cout<<"The prime numbers upto "<<n<<" are: "<<endl;
    for(int i=2;i<=n;i++)
    {
        count=0;
        for(int j=2;j<i;j++)
        {
            if(i%j==0)
            {
                count++;
            }
        }
        if(count==0)
        {
            cout<<i<<endl;
        }
    }
}
```

LAB 2**Output:**

```
C:\jonash chataut\lab 2\WAP  X + | ~
Enter the value upto which you want the prime numbers:
50
The prime numbers upto 50 are:
2
3
5
7
11
13
17
19
23
29
31
37
41
43
47
-----
Process exited after 10.05 seconds with return value 0
Press any key to continue . . . |
```

2. WAP to perform arithmetic operation on two numbers using switch case

```
#include<iostream>
using namespace std;
int main(){
    float a,b,r;
    char c;
    cout<<"Input two number: "<<endl;
    cin>>a>>b;
    cout<<"Enter the operator you want to do: "<<endl;
    cin>>c;
    switch(c)
    {
        case '+':
            r=a+b;
            cout<<"The result is "<<a<<"+"<<b<<"="<<r;
            break;
```

LAB 2

```
case '-':  
    r=a-b;  
    cout<<"The result is "<<a<<"-"<<b<<"="<<r;  
    break;  
  
case '*':  
    r=a*b;  
    cout<<"The result is "<<a<<"*"<<b<<"="<<r;  
    break;  
  
case '/':  
    r=a/b;  
    cout<<"The result is "<<a<<"/"<<b<<"="<<r;  
    break;  
default:  
    cout<<"Error"<<endl;  
    break;  
}  
return 0;  
}
```

Output:

```
C:\jonash chataut\lab 2\switch > + |  
Input two number:  
4  
5  
Enter the operator you want to do:  
*  
The result is 4*5=20  
-----  
Process exited after 9.888 seconds with return value 0  
Press any key to continue . . . |
```

LAB 2**3. WAP to swap two numbers using**

- a) Call by value
- b) Call by reference
- c) Call by pointer

a) Call by value

```
#include<iostream>
using namespace std;
void swap(int a, int b)
{
    int c;
    cout<<"Before swaping:"<<a<<" "<<b<<endl;
    c=a;
    a=b;
    b=c;
    cout<<"After swaping:"<<a<<" "<<b<<endl;
}
int main(){
    int n1,n2;
    cout<<"Enter two numbers: "<<endl;
    cin>>n1>>n2;
    swap(n1,n2);
    return 0;
}
```

Output:

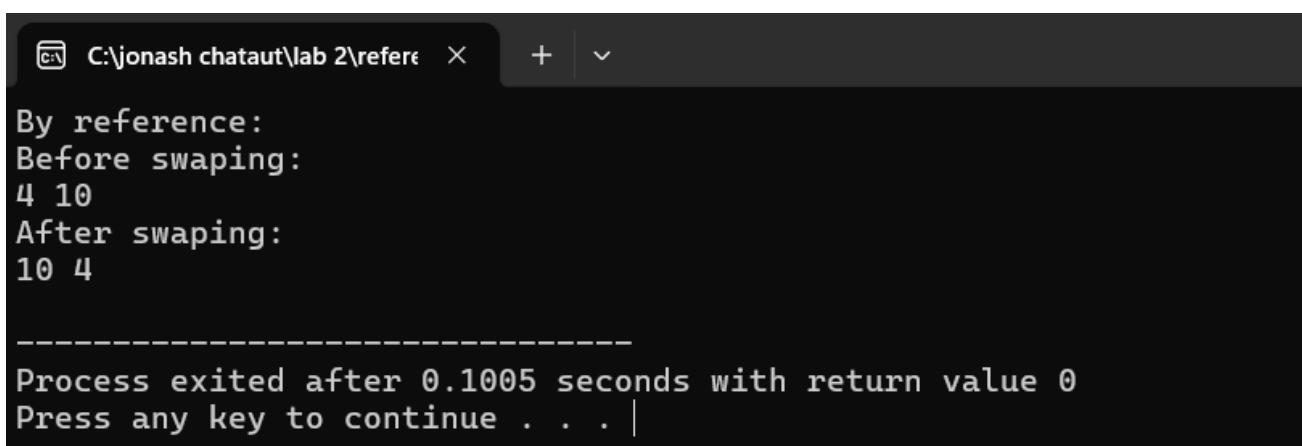
```
C:\jonash chataut\lab 2\swapi > + | 
Enter two numbers:
4
10
Before swaping:4 10
After swaping:10 4
-----
Process exited after 7.508 seconds with return value 0
Press any key to continue . . . |
```

LAB 2

- b) Call by reference

```
#include<iostream>
using namespace std;
void swapbyreference(int &a, int &b)
{
    int temp=a;
    a=b;
    b=temp;
}

int main()
{
    int n1=4,n2=10;
    cout<<"By reference:"<<endl;
    cout<<"Before swaping:"<<endl;
    cout<<n1<<" "<<n2<<endl;
    swapbyreference(n1,n2);
    cout<<"After swaping:"<<endl;
    cout<<n1<<" "<<n2<<endl;
    return 0;
}
```

Output:

```
C:\jonash chataut\lab 2\referen X + ▾
By reference:
Before swaping:
4 10
After swaping:
10 4

-----
Process exited after 0.1005 seconds with return value 0
Press any key to continue . . . |
```

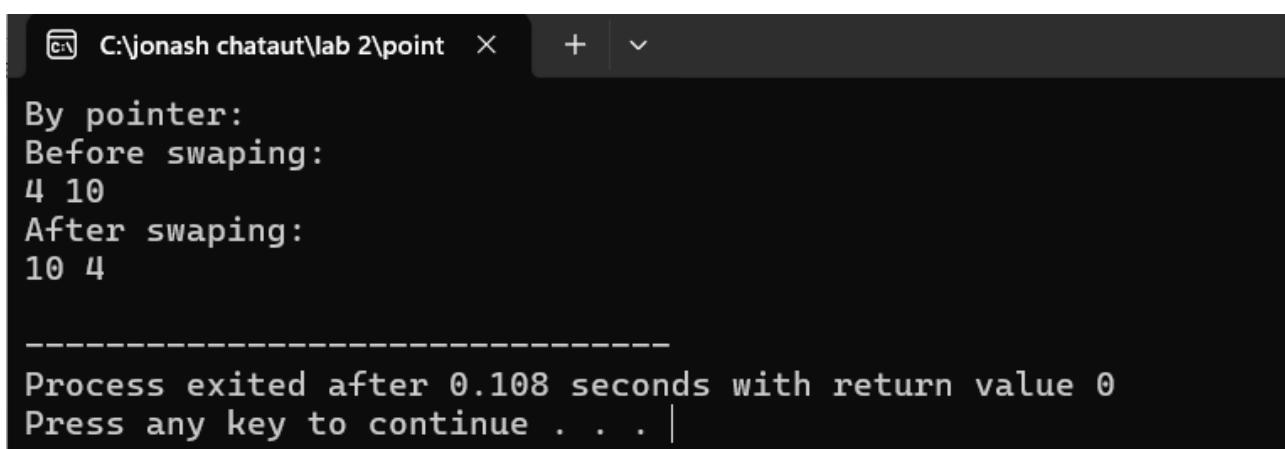
LAB 2

c) Call by pointer

```
#include<iostream>
using namespace std;
void swapbyreference(int *a, int *b)
{
    int temp=*a;
    *a=*b;
    *b=temp;
}

int main()
{
    int n1=4,n2=10;
    cout<<"By pointer:"<<endl;
    cout<<"Before swaping:"<<endl;
    cout<<n1<<" "<<n2<<endl;
    swapbyreference(&n1,&n2);
    cout<<"After swaping:"<<endl;
    cout<<n1<<" "<<n2<<endl;
    return 0;
}
```

Output:



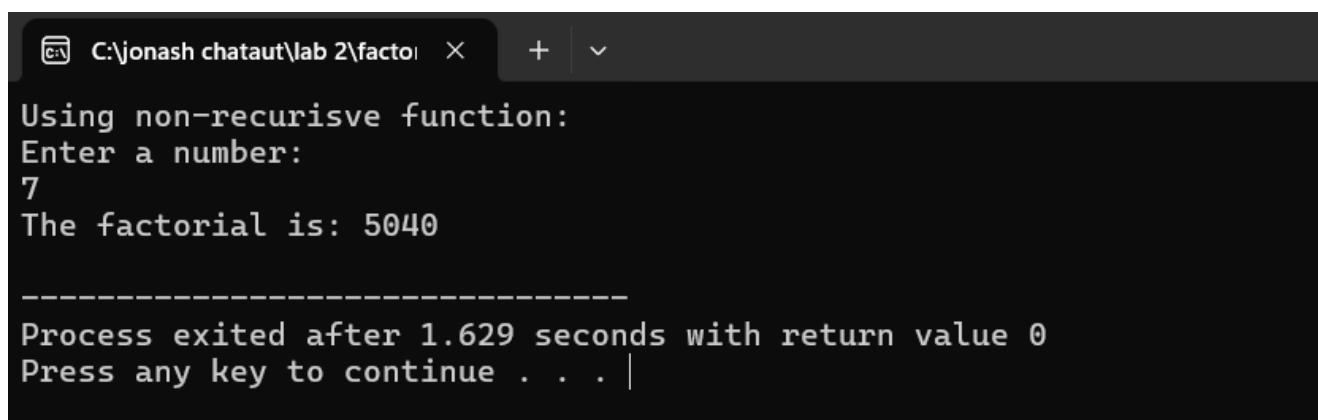
```
C:\jonash chataut\lab 2\point  X  +  ▾
By pointer:
Before swaping:
4 10
After swaping:
10 4

-----
Process exited after 0.108 seconds with return value 0
Press any key to continue . . . |
```

LAB 2**4. WAP to find factorial of a number using**

a) non-recursive function

```
#include<iostream>
using namespace std;
int facto(int n)
{
    int fac=1;
    for(int i=1;i<=n;i++)
    {
        fac=fac*i;
    }
    return fac;
}
int main()
{
    int num;
    cout<<"Using non-recursve function:"<<endl;
    cout<<"Enter a number:"<<endl;
    cin>>num;
    cout<<"The factorial is: "<<facto(num)<<endl;
    return 0;
}
```

Output:

The screenshot shows a terminal window with the following text output:

```
C:\jonash chataut\lab 2\facto X + | 
Using non-recursve function:
Enter a number:
7
The factorial is: 5040
-----
Process exited after 1.629 seconds with return value 0
Press any key to continue . . . |
```

LAB 2

- b) recursive function

```
#include<iostream>

using namespace std;

int facto(int n)

{

    int fac;

    if(n<=1)

    {

        return 1;

    }

    else

        return n*facto(n-1);

}

int main()

{

    int num;

    cout<<"Using recursive function:"<<endl;

    cout<<"Enter a number:"<<endl;

    cin>>num;

    cout<<"The factorial is: "<<facto(num)<<endl;

    return 0;

}
```

Output:

```
C:\jonash chataut\lab 2\facto × + | ▾

Using recursive function:
Enter a number:
5
The factorial is: 120

-----
Process exited after 4.934 seconds with return value 0
Press any key to continue . . . |
```

LAB 2

5. WAP and create namespace with different function and data. Perform following

a) scope resolution to access data and function

b) using keyword

a) scope resolution to access data and function

```
#include<iostream>
```

```
using namespace std;
```

```
namespace jon
```

```
{
```

```
    int n=17;
```

```
    int add(int a, int b)
```

```
{
```

```
    return a+b;
```

```
}
```

```
    int sub(int a, int b)
```

```
{
```

```
    return a-b;
```

```
}
```

```
}
```

```
int main(){
```

```
    cout<<"Using scope resolution"<<endl;
```

```
    cout<<"n from namespace jon "<<jon::n<<endl;
```

```
    cout<<"adding 2 and 4 "<<jon::add(2,4)<<endl;
```

```
    cout<<"subtracting 7 from 8 "<<jon::sub(8,7)<<endl;
```

```
    return 0;
```

```
}
```

LAB 2**Output:**

```
C:\jonash chataut\lab 2\using  X + | v

Using scope resolution
n from namespace jon 17
adding 2 and 4 6
subtracting 7 from 8 1

-----
Process exited after 0.1366 seconds with return value 0
Press any key to continue . . . |
```

b) using keyword

```
#include<iostream>

using namespace std;

namespace jon {

    int n=17;

    int add(int a, int b)

    {

        return a+b;

    }

    int sub(int a, int b)

    {

        return a-b;

    }

}

using namespace jon;

int main(){

    cout<<"Using keyword"<<endl;
    cout<<"n from namespace jon "<< n <<endl;
    cout<<"adding 2 and 4 "<<add(2,4)<<endl;
    cout<<"subtracting 7 from 8 "<<sub(8,7)<<endl;
    return 0;
}
```

LAB 2

Output:

```
C:\jonash chataut\lab 2\name  X + ▾

Using keyword
n from namespace jon 17
adding 2 and 4 6
substracting 7 from 8 1

-----
Process exited after 0.1436 seconds with return value 0
Press any key to continue . . . |
```

6. WAP to find sum of two number using inline function

```
#include<iostream>
using namespace std;
inline add(int a, int b)
{
    return a+b;
}
int main()
{
    int a,b;
    cout<<"Enter two numbers: "<<endl;
    cin>>a>>b;
    cout<<"Sum is: "<<add(a,b);
    return 0;
}
```

Output:

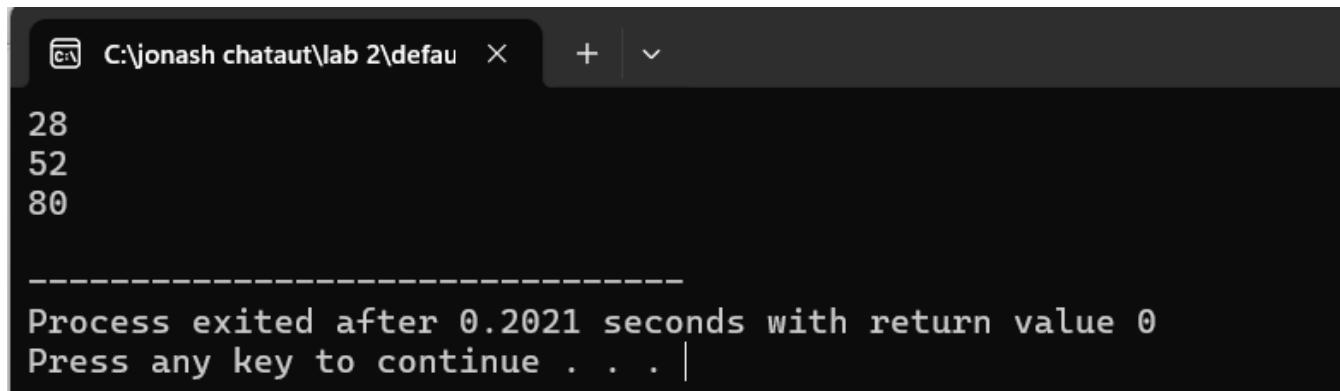
```
C:\jonash chataut\lab 2\inline  X + ▾

Enter two numbers:
4
5
Sum is: 9
-----
Process exited after 2.614 seconds with return value 0
Press any key to continue . . . |
```

LAB 2**7. WAP to illustrate use of default argument**

```
#include<iostream>
using namespace std;
int sum(int x, int y, int z = 1, int w = 2)
{
    return (x + y + z + w);
}

int main()
{
    cout << sum(10, 15) << endl;
    cout << sum(10, 15, 25) << endl;
    cout << sum(10, 15, 25, 30) << endl;
    return 0;
}
```

Output:

```
28
52
80
-----
Process exited after 0.2021 seconds with return value 0
Press any key to continue . . . |
```

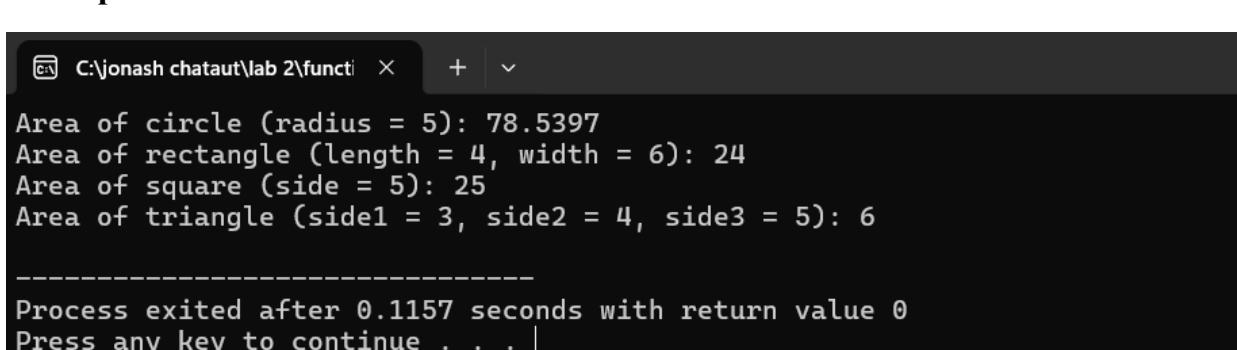
8. WAP to overload area function to calculate area of triangle, rectangle, square and circle

```
#include <iostream>
#include<cmath>

const float PI = 3.14159;
using namespace std;
```

LAB 2

```
float area(float radius) {  
    return PI * radius * radius;  
}  
  
float area(float length, float width) {  
    return length * width;  
}  
  
float area_sq(float side) {  
    return side * side;  
}  
  
float area(float side1, float side2, float side3) {  
    float s = (side1 + side2 + side3) / 2;  
    return sqrt(s * (s - side1) * (s - side2) * (s - side3));  
}  
  
int main() {  
    cout << "Area of circle (radius = 5): " << area(5) << endl;  
    cout << "Area of rectangle (length = 4, width = 6): " << area(4, 6) << endl;  
    cout << "Area of square (side = 5): " << area_sq(5) << endl;  
    cout << "Area of triangle (side1 = 3, side2 = 4, side3 = 5): " << area(3, 4, 5) <<  
    endl;  
    return 0;  
}
```

Output:

```
C:\jonash chataut\lab 2\functi  
Area of circle (radius = 5): 78.5397  
Area of rectangle (length = 4, width = 6): 24  
Area of square (side = 5): 25  
Area of triangle (side1 = 3, side2 = 4, side3 = 5): 6  
-----  
Process exited after 0.1157 seconds with return value 0  
Press any key to continue . . . |
```

LAB 2

8. WAP to overloaded function to find area of triangle with three sides, square and circle. Note for triangle take one side fixed value of 5

```
#include <iostream>
#include<cmath>
const float PI = 3.14159;
using namespace std;

float area(float radius) {
    return PI * radius * radius;
}

float area_sq(float side) {
    return side * side;
}

float area(float s1, float s2, float s3=5) {
    float s = (s1 + s2 + s3) / 2;
    return sqrt(s * (s - s1) * (s - s2) * (s - s3));
}

int main() {
    int r,l,s1,s2;
    cout<<"Enter the radius of the circle: "<<endl;
    cin>>r;
    cout<<"Enter the length of the side of square: "<<endl;
    cin>>l;
    cout<<"Enter the value of remaining two sides of triangle one side is fixed at 5:"<<endl;
    cin>>s1>>s2;
    cout << "Area of circle: " << area(r) << endl;
    cout << "Area of square: " << area_sq(l) << endl;
    cout << "Area of triangle: " << area(s1,s2) << endl;
    return 0;
}
```

LAB 2

Output:

```
C:\jonash chataut\lab 2\one s  X  +  ▾  
Enter the radius of the circle:  
2  
Enter the length of the side of square:  
4  
Enter the value of remaining two sides of triangle one side is fixed at 5:  
7  
8  
Area of circle: 12.5664  
Area of square: 16  
Area of triangle: 17.3205  
-----  
Process exited after 8.561 seconds with return value 0  
Press any key to continue . . . |
```

LAB 3

1. WAP to pass array in function to find maximum element of the array entered by user using

- a) pointer as parameter**
- b) sized array as parameter**
- c) unsized array as parameter**

- a) pointer as parameter**

```
#include<iostream>
using namespace std;
int array(int *ptr)
{
    int lar;
    lar=*ptr;
    for (int i=0;i<5;i++)
    {
        if(*(ptr+i)>lar)
        {
            lar=*(ptr+i);
        }
    }
    return lar;
}

int main()
{
    int ar[5];
    cout<<"Enter the elements of array"<<endl;
    for(int i=0;i<5;i++)
    {
        cin>>ar[i];
    }
    cout<<"The largest element of the array is "<<array(ar)<<endl;
    return 0;
}
```

Output:

```
C:\Jonash oop\lab 3\pointer a > + ->
Enter the elements of array
4
5
8
1
7
The largest element of the array is 8
-----
Process exited after 4.284 seconds with return value 0
Press any key to continue . . . |
```

LAB 3

b) sized array as parameter

```
#include<iostream>
using namespace std;
int array(int ary[5])
{
    int lar;
    lar=ary[0];
    for (int i=0;i<5;i++)
    {
        if(ary[i]>lar)
        {
            lar=ary[i];
        }
    }
    return lar;
}

int main()
{
    int ar[5];
    cout<<"Enter the elements of array"<<endl;
    for(int i=0;i<5;i++)
    {
        cin>>ar[i];
    }
    cout<<"The largest element of the array is "<<array(ar)<<endl;
    return 0;
}
```

Output:

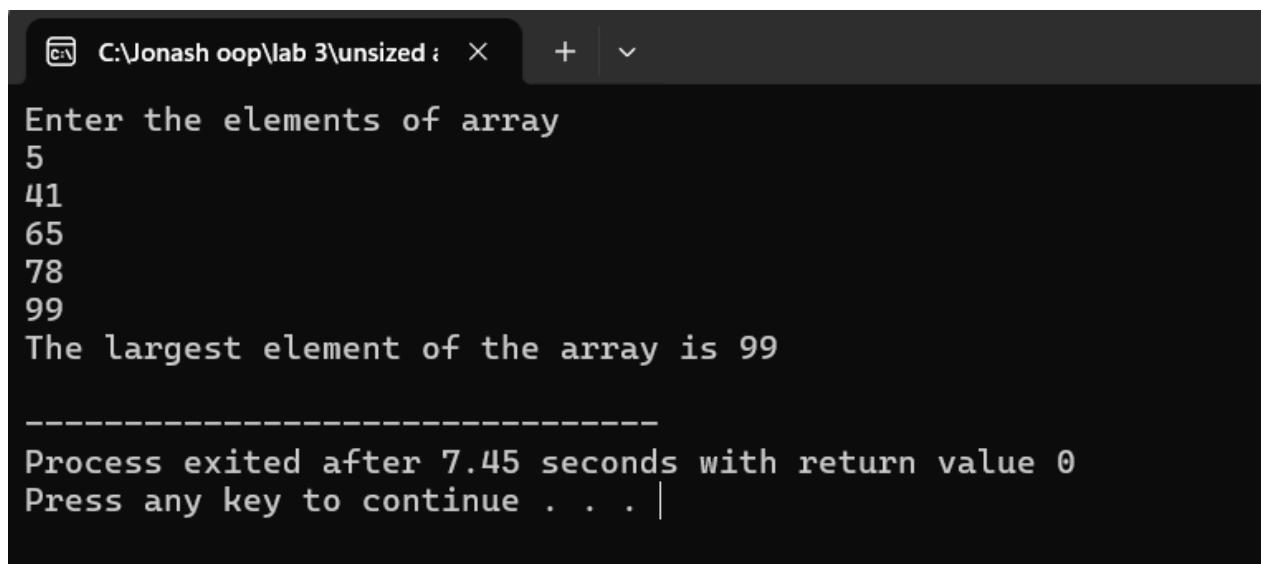
```
C:\Jonash oop\lab 3\sized arr > Enter the elements of array
4
5
9
8
2
The largest element of the array is 9
-----
Process exited after 5.917 seconds with return value 0
Press any key to continue . . . |
```

LAB 3

```
c) unsized array as parameter
#include<iostream>
using namespace std;
int array(int ary[])
{
    int lar;
    lar=*ary;
    for (int i=0;i<5;i++)
    {
        if(ary[i]>lar)
        {
            lar=ary[i];
        }
    }
    return lar;
}

int main ()
{
    int ar[5];
    cout<<"Enter the elements of array"<<endl;
    for (int i=0;i<5;i++)
    {
        cin>>ar[i];
    }
    cout<<"The largest element of the array is "<<array(ar)<<endl;
    return 0;
}
```

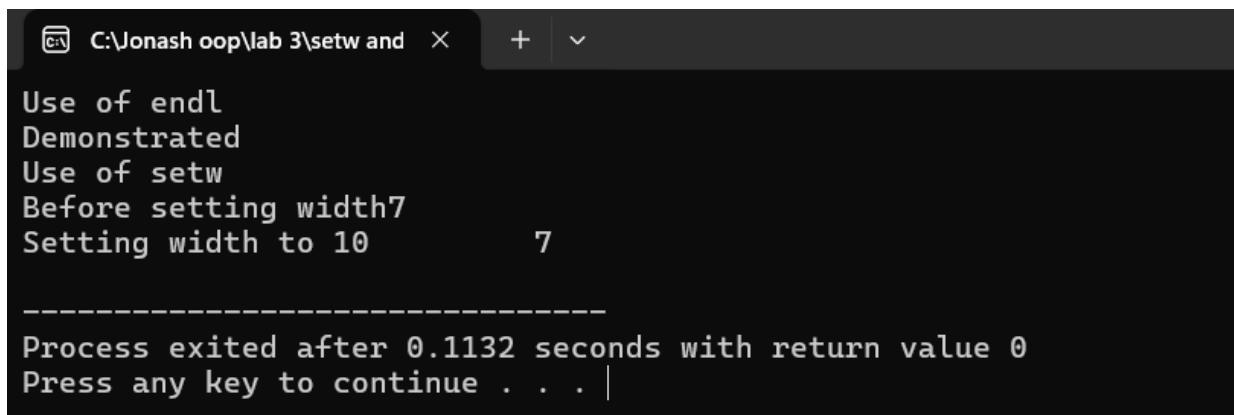
Output:



```
C:\Jonash oop\lab 3\unsized > Enter the elements of array
5
41
65
78
99
The largest element of the array is 99
-----
Process exited after 7.45 seconds with return value 0
Press any key to continue . . . |
```

LAB 3**2. WAP to demonstrate use of endl and setw manipulator.**

```
#include<iostream>
#include<iomanip>
using namespace std;
int main()
{
    int num=7;
    cout<<"Use of endl"<<endl<<"Demonstrated"<<endl;
    cout<<"Use of setw"<<endl;
    cout<<"Before setting width"<<num<<endl;
    cout<<"Setting width to 10"<<setw(10)<<num<<endl;
    return 0;
}
```

Output:

```
C:\Jonash oop\lab 3\setw and + ▾
Use of endl
Demonstrated
Use of setw
Before setting width7
Setting width to 10      7
-----
Process exited after 0.1132 seconds with return value 0
Press any key to continue . . . |
```

3. WAP to show type concusion implicit and explicit.

```
#include<iostream>
using namespace std;
int main ()
{
    int a,b;
    float r;
    a=7;
    b=4;
    cout<<"Dividing 7 by 4 in both cases"<<endl;
    cout<<"Implicit case:"<<endl;
    r=a/b;
    cout<<r<<endl;
    cout<<"Explicit case:"<<endl;
    r=float(a)/b;
    cout<<r<<endl;
    return 0;
}
```

LAB 3

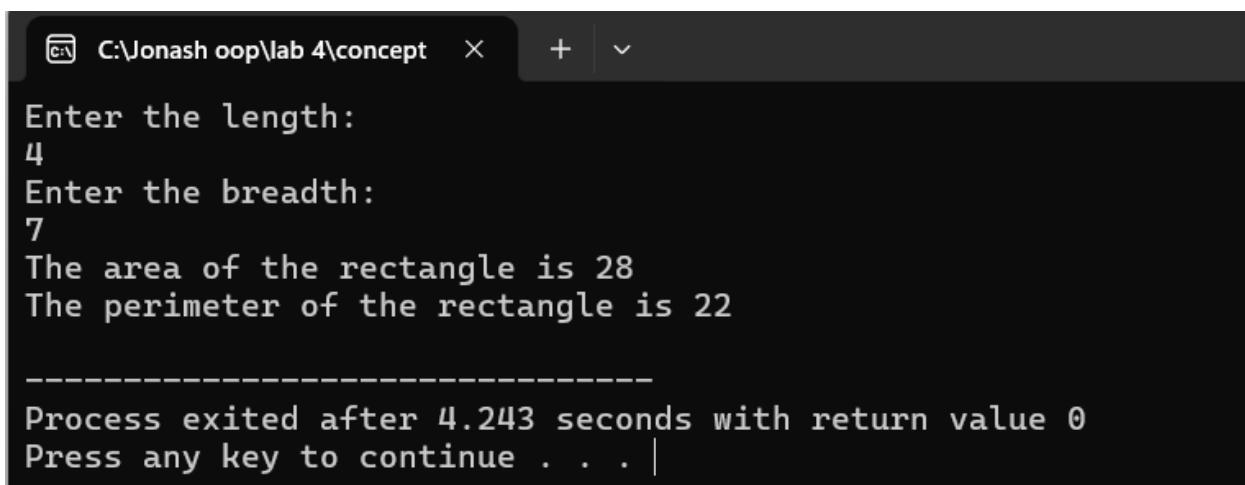
Output:

```
C:\Jonash oop\lab 3\implicit & X + ▾  
Dividing 7 by 4 in both cases  
Implicit case:  
1  
Explicit case:  
1.75  
-----  
Process exited after 0.1065 seconds with return value 0  
Press any key to continue . . . |
```

LAB 4**1. WAP to find the area and perimeter of rectangle usnig concept of class and object.**

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

class rectangle
{
    int l,b;
public:
    void get()
    {
        cout<<"Enter the length:"<<endl;
        cin>>l;
        cout<<"Enter the breadth:"<<endl;
        cin>>b;
    }
    int area()
    {
        return l*b;
    }
    int perimeter()
    {
        return 2*(l+b);
    }
};
int main()
{
    rectangle r;
    r.get();
    cout<<"The area of the rectangle is "<<r.area()<<endl;
    cout<<"The perimeter of the rectangle is "<<r.perimeter()<<endl;
    return 0;
}
```

Output:

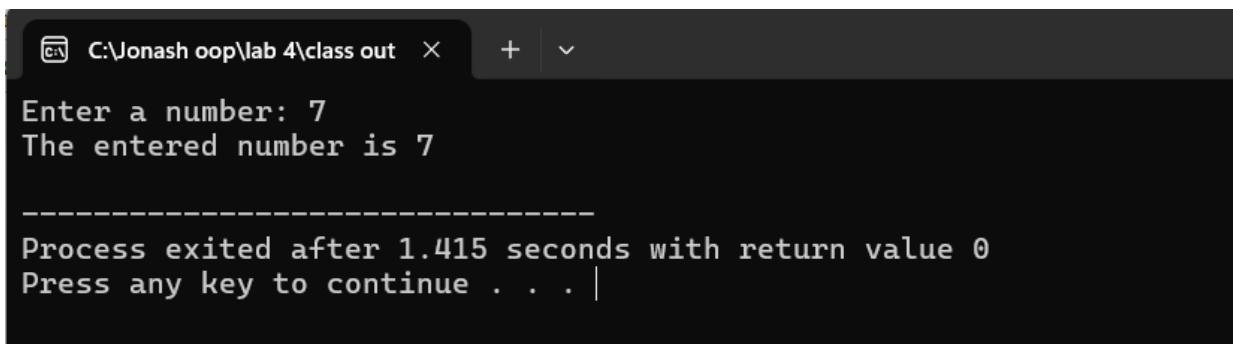
```
C:\Jonash oop\lab 4\concept  X  +  ▾
Enter the length:
4
Enter the breadth:
7
The area of the rectangle is 28
The perimeter of the rectangle is 22
-----
Process exited after 4.243 seconds with return value 0
Press any key to continue . . . |
```

LAB 4**2. WAP to set and show data by using concept of class. Define member function outside class.**

```
#include<iostream>
using namespace std;
class concept
{
    int x;
public:
    void getdata();
};

void concept::getdata()
{
    cout<<"Enter a number: ";
    cin>>x;
    cout<<"The entered number is "<<x<<endl;
}

int main ()
{
    concept a;
    a.getdata();
    return 0;
}
```

Output:

```
C:\Jonash oop\lab 4\class out + ▾
Enter a number: 7
The entered number is 7
-----
Process exited after 1.415 seconds with return value 0
Press any key to continue . . . |
```

LAB 4**3. WAP designing a class to represent item information. Include following member**

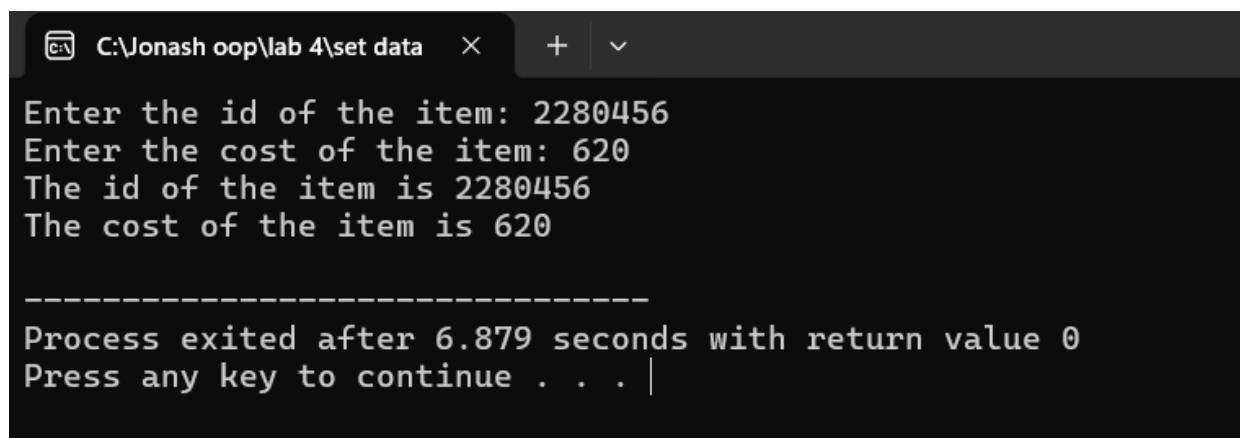
-Data members -Member function

Itemid Set data

Cost Show data

```
#include<iostream>
using namespace std;
class item
{
    int itemid,cost;
public:
    void setdata();
    void showdata();
};
void item::setdata()
{
    cout<<"Enter the id of the item: ";
    cin>>item::itemid;
    cout<<"Enter the cost of the item: ";
    cin>>item::cost;
}
void item::showdata()
{
    cout<<"The id of the item is "<<item::itemid<<endl;
    cout<<"The cost of the item is "<<item::cost<<endl;
}

int main()
{
    item a;
    a.setdata();
    a.showdata();
    return 0;
}
```

Output:

```
C:\Jonash oop\lab 4\set data  X  +  ▾

Enter the id of the item: 2280456
Enter the cost of the item: 620
The id of the item is 2280456
The cost of the item is 620

-----
Process exited after 6.879 seconds with return value 0
Press any key to continue . . . |
```

LAB 4

4. WAP to find sum of two complex number entered by user using concept of passing object as argument.[Use pass by value]

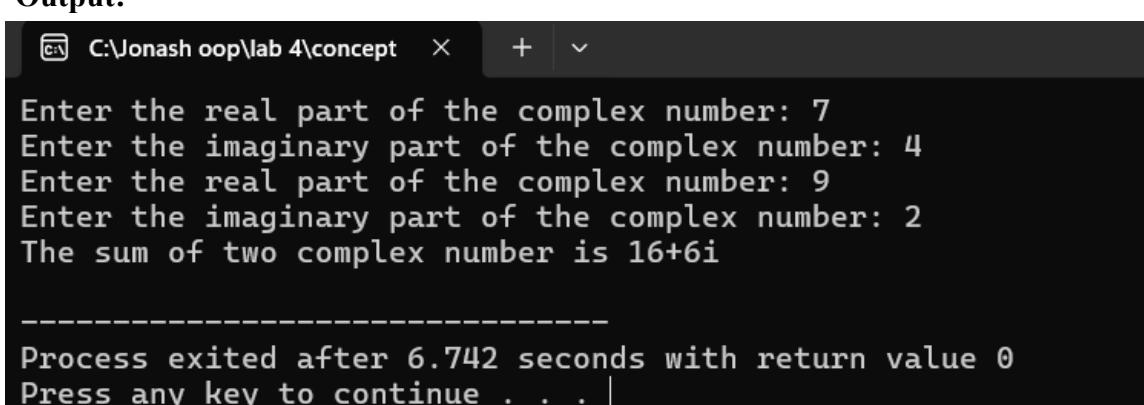
```
#include<iostream>
using namespace std;
class complex
{
    int r,i;
public:
    void get()
    {
        cout<<"Enter the real part of the complex number: ";
        cin>>r;
        cout<<"Enter the imaginary part of the complex number: ";
        cin>>i;
    }

    void sum(complex x, complex y)
    {
        r=x.r+y.r;
        i=x.i+y.i;
    }

    void show()
    {
        cout<<"The sum of two complex number is
" <<r << "+" <<i <<"i" << endl;
    }
};

int main()
{
    complex a,b,c;
    a.get();
    b.get();
    c.sum(a,b);
    c.show();
    return 0;
}
```

Output:



```
C:\Jonash oop\lab 4\concept + | v
Enter the real part of the complex number: 7
Enter the imaginary part of the complex number: 4
Enter the real part of the complex number: 9
Enter the imaginary part of the complex number: 2
The sum of two complex number is 16+6i
-----
Process exited after 6.742 seconds with return value 0
Press any key to continue . . . |
```

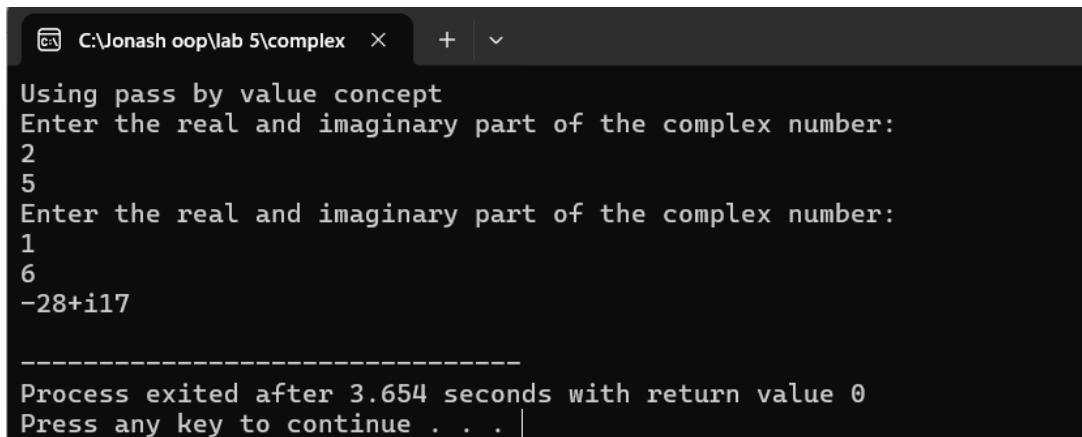
LAB 5**1. WAP to multiply two complex numbers using the concept of pass by value taking object as an argument.**

```
#include<iostream>
using namespace std;
class complex
{
    int r,i;
public:
    void get()
    {
        cout<<"Enter the real and imaginary part of the complex number:"<<endl;
        cin>>r>>i;
    }

    void show()
    {
        cout<<r<<"+"<<"i"<<i<<endl;
    }

    void multiply(complex x, complex y)
    {
        r=x.r*y.r-x.i*y.i;
        i=x.r*y.i+x.i*y.r;
    }
};

int main()
{
    cout<<"Using pass by value concept"<<endl;
    complex a,b,c;
    a.get();
    b.get();
    c.multiply(a,b);
    c.show();
    return 0;
}
```

Output:

```
C:\Jonash oop\lab 5\complex  x  +  v

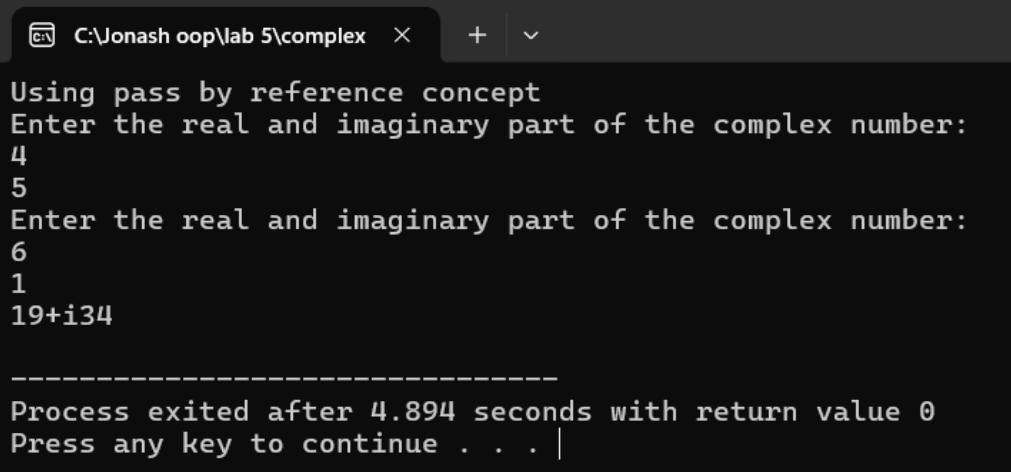
Using pass by value concept
Enter the real and imaginary part of the complex number:
2
5
Enter the real and imaginary part of the complex number:
1
6
-28+i17

-----
Process exited after 3.654 seconds with return value 0
Press any key to continue . . . |
```

LAB 5**2. WAP to multiply two complex numbers using the concept of pass by reference taking object as an argument.**

```
#include<iostream>
using namespace std;
class complex
{
    int r,i;
public:
    void get()
    {
        cout<<"Enter the real and imaginary part of the complex number:"<<endl;
        cin>>r>>i;
    }
    void show()
    {
        cout<<r<<"+"<<"i"<<i<<endl;
    }
    void multiply(complex &x, complex &y)
    {
        r=x.r*y.r-x.i*y.i;
        i=x.r*y.i+x.i*y.r;
    }
};

int main()
{
    cout<<"Using pass by reference concept"<<endl;
    complex a,b,c;
    a.get();
    b.get();
    c.multiply(a,b);
    c.show();
    return 0;
}
```

Output:

```
C:\Jonash oop\lab 5\complex  +  ~
Using pass by reference concept
Enter the real and imaginary part of the complex number:
4
5
Enter the real and imaginary part of the complex number:
6
1
19+i34
-----
Process exited after 4.894 seconds with return value 0
Press any key to continue . . . |
```

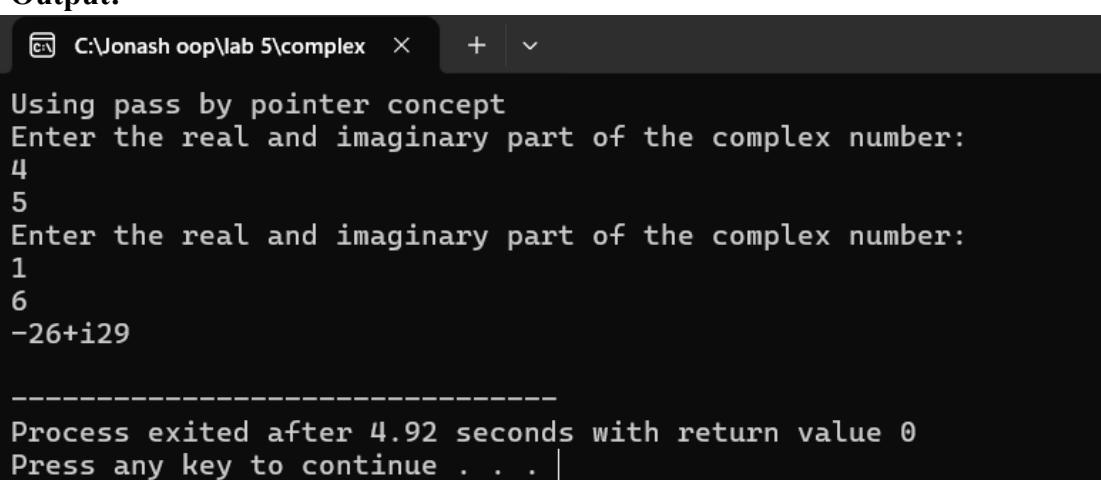
LAB 5**3. WAP to multiply two complex numbers using the concept of pass by pointer taking object as an argument.**

```
#include<iostream>
using namespace std;
class complex
{
    int r,i;
public:
    void get()
    {
        cout<<"Enter the real and imaginary part of the complex number:"<<endl;
        cin>>r>>i;
    }

    void show()
    {
        cout<<r<<"+"<<"i"<<i<<endl;
    }

    void multiply(complex *x, complex *y)
    {
        r=x->r*y->r-x->i*y->i;
        i=x->r*y->i+x->i*y->r;
    }
};

int main()
{
    cout<<"Using pass by pointer concept"<<endl;
    complex a,b,c;
    a.get();
    b.get();
    c.multiply(&a,&b);
    c.show();
    return 0;
}
```

Output:

```
C:\Jonash oop\lab 5\complex  x  +  v

Using pass by pointer concept
Enter the real and imaginary part of the complex number:
4
5
Enter the real and imaginary part of the complex number:
1
6
-26+i29

-----
Process exited after 4.92 seconds with return value 0
Press any key to continue . . . |
```