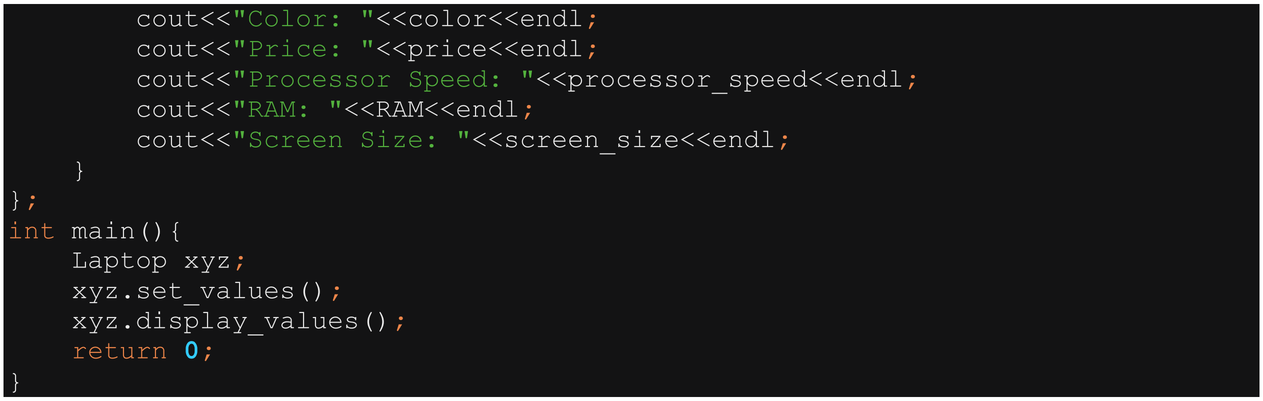
Lab Manual:-

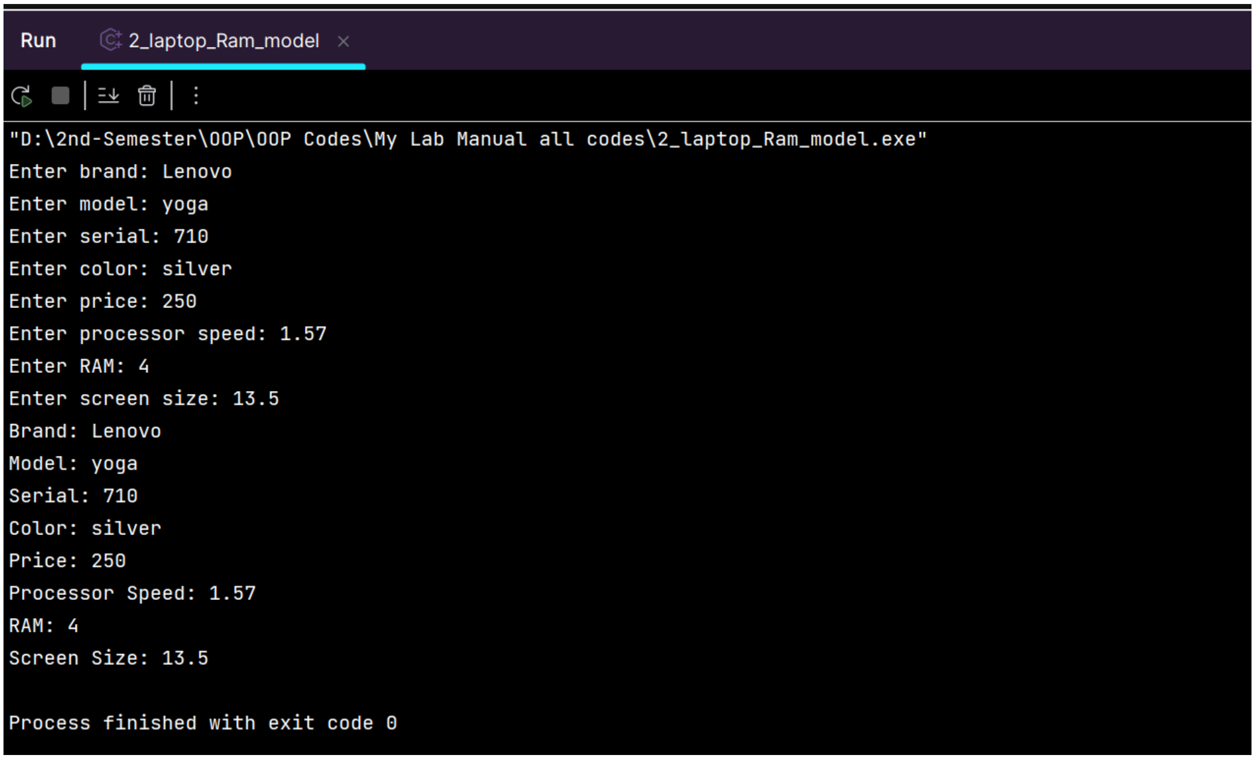
Code no 1:-

```
//
// Created by zohaib on 29/05/2023.
//
/*Q#.1: Write a C++ program that creates a class called laptop. The data
members of the class
        are brand (string), model (string), serial (int), color (string),
price (float),
        processor speed (float), RAM (int), screen size(float). Create member
function
        that will set the individual values. Since the RAM can be upgraded
therefore create
        a function that allows you to upgrade the RAM only. In the end,
create a function
        that will display all the data members.*/

#include<iostream>
using namespace std;

class Laptop{
private:
    string brand;
    string model;
    string serial;
    string color;
    float price;
    float processor_speed;
    int RAM;
    float screen_size;
public:
    void set_values(){
        cout<<"Enter brand: ";
        cin>>brand;
        cout<<"Enter model: ";
        cin>>model;
        cout<<"Enter serial: ";
        cin>>serial;
        cout<<"Enter color: ";
        cin>>color;
        cout<<"Enter price: ";
        cin>>price;
        cout<<"Enter processor speed: ";
        cin>>processor_speed;
        cout<<"Enter RAM: ";
        cin>>RAM;
        cout<<"Enter screen size: ";
        cin>>screen_size;
    }
    void display_values(){
        cout<<"Brand: "<<brand<<endl;
        cout<<"Model: "<<model<<endl;
        cout<<"Serial: "<<serial<<endl;
```



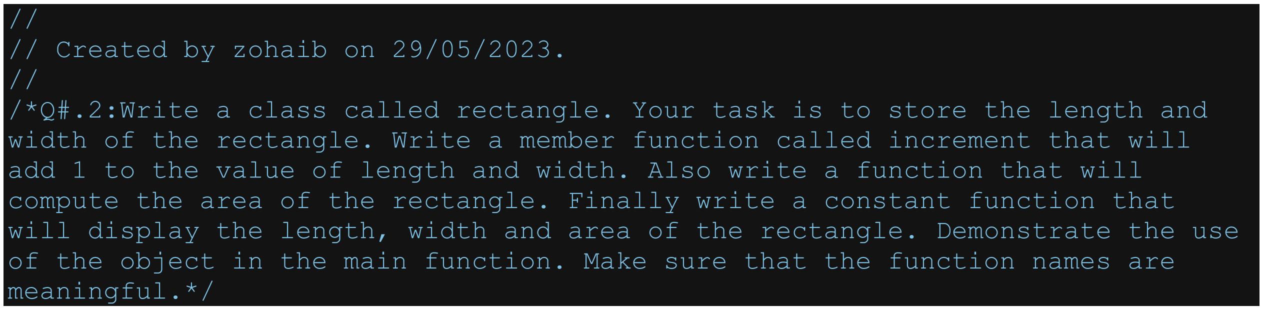
```
        cout<<"Color: "<<color<<endl;
        cout<<"Price: "<<price<<endl;
        cout<<"Processor Speed: "<<processor_speed<<endl;
        cout<<"RAM: "<<RAM<<endl;
        cout<<"Screen Size: "<<screen_size<<endl;
    }
};
int main(){
    Laptop xyz;
    xyz.set_values();
    xyz.display_values();
    return 0;
}
```

Output:-

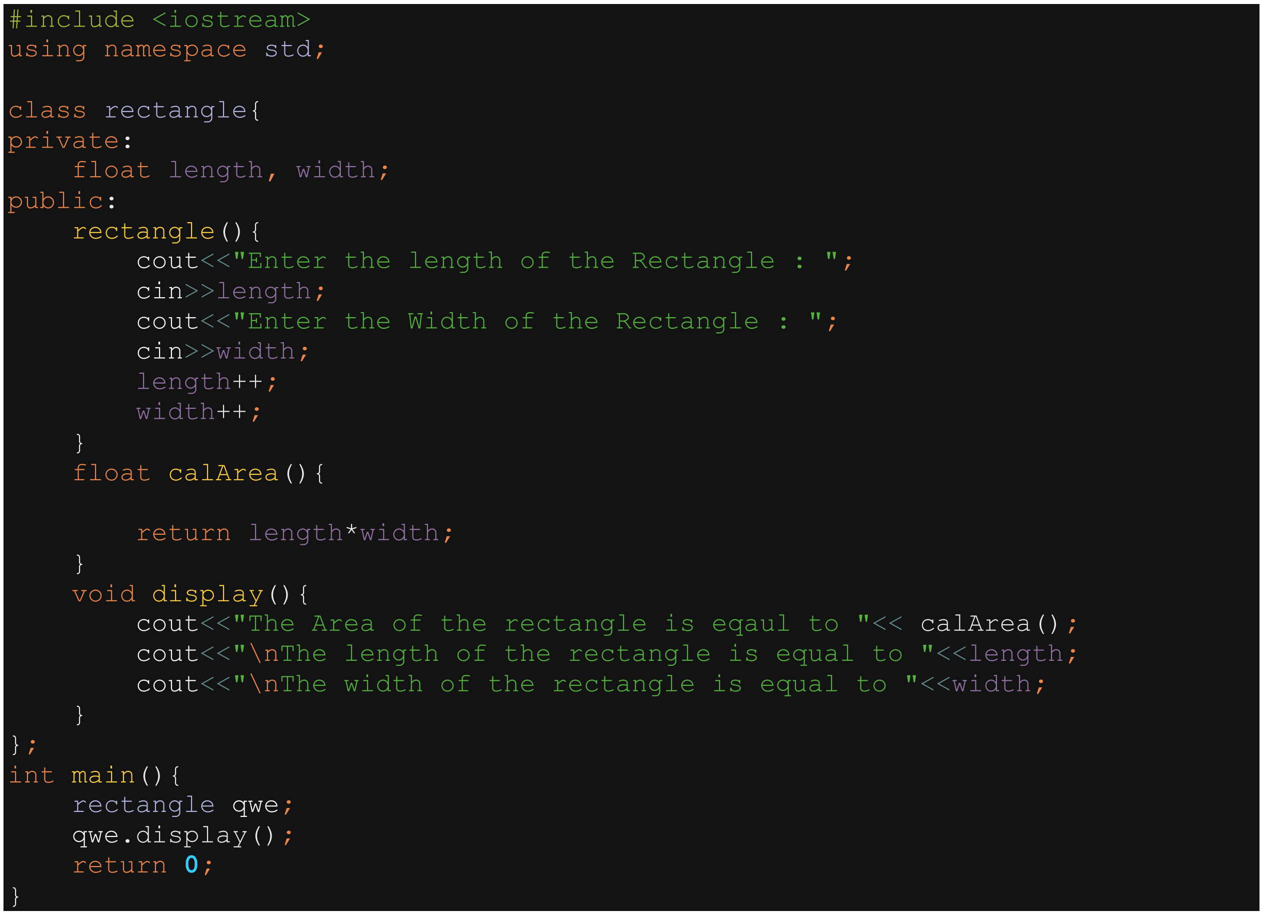


```
"D:\2nd-Semester\OOP\OOP Codes\My Lab Manual all codes\2_laptop_Ram_model.exe"
Enter brand: Lenovo
Enter model: yoga
Enter serial: 710
Enter color: silver
Enter price: 250
Enter processor speed: 1.57
Enter RAM: 4
Enter screen size: 13.5
Brand: Lenovo
Model: yoga
Serial: 710
Color: silver
Price: 250
Processor Speed: 1.57
RAM: 4
Screen Size: 13.5

Process finished with exit code 0
```

Code no 2:-

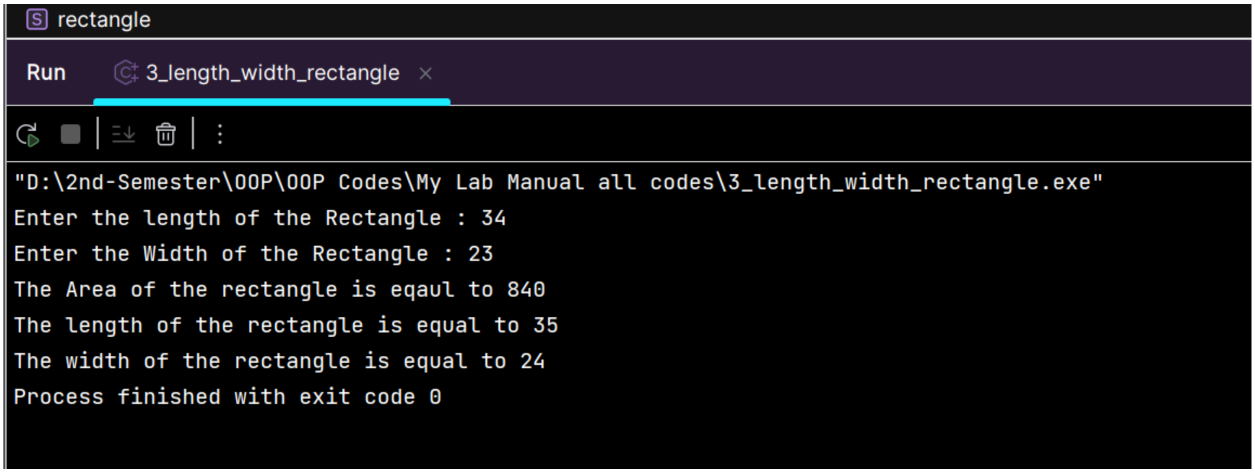


```
//
// Created by zohaib on 29/05/2023.
//
/*Q#.2:Write a class called rectangle. Your task is to store the length and
width of the rectangle. Write a member function called increment that will
add 1 to the value of length and width. Also write a function that will
compute the area of the rectangle. Finally write a constant function that
will display the length, width and area of the rectangle. Demonstrate the use
of the object in the main function. Make sure that the function names are
meaningful.*/
```



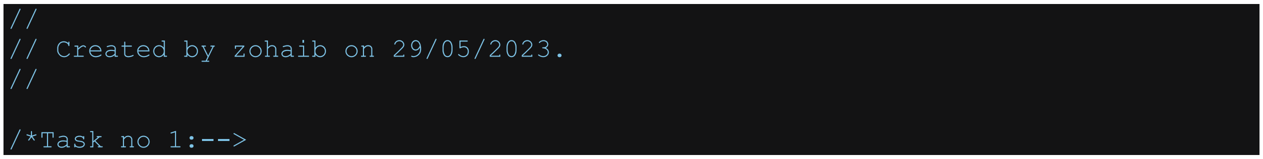
```cpp
#include <iostream>
using namespace std;

class rectangle{
private:
    float length, width;
public:
    rectangle(){
        cout<<"Enter the length of the Rectangle : ";
        cin>>length;
        cout<<"Enter the Width of the Rectangle : ";
        cin>>width;
        length++;
        width++;
    }
    float calArea(){

        return length*width;
    }
    void display(){
        cout<<"The Area of the rectangle is eqaul to "<< calArea();
        cout<<"\nThe length of the rectangle is equal to "<<length;
        cout<<"\nThe width of the rectangle is equal to "<<width;
    }
};
int main(){
    rectangle qwe;
    qwe.display();
    return 0;
}
```

Output:-

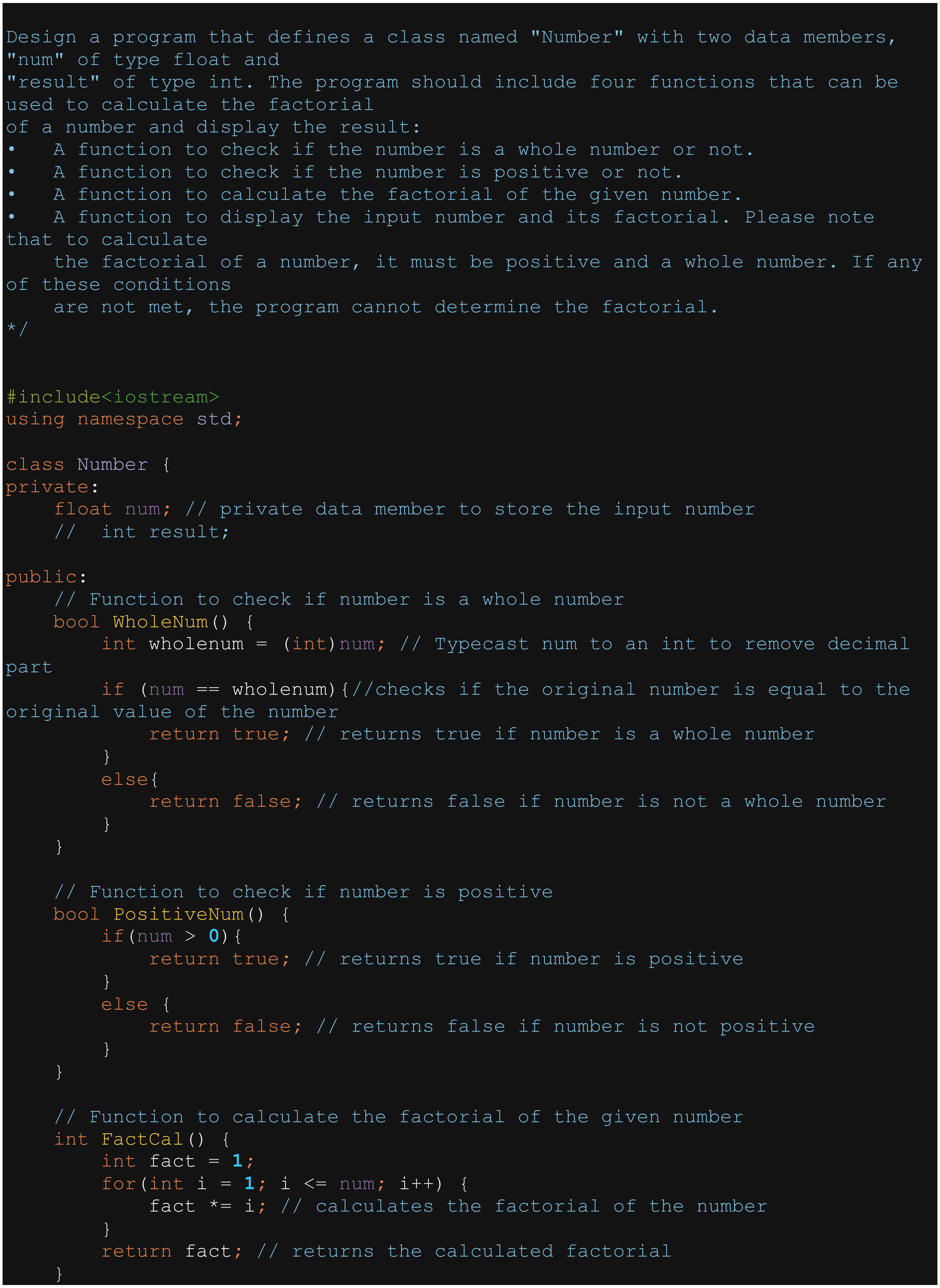


```
"D:\2nd-Semester\OOP\OOP Codes\My Lab Manual all codes\3_length_width_rectangle.exe"
Enter the length of the Rectangle : 34
Enter the Width of the Rectangle : 23
The Area of the rectangle is eqaul to 840
The length of the rectangle is equal to 35
The width of the rectangle is equal to 24
Process finished with exit code 0
```

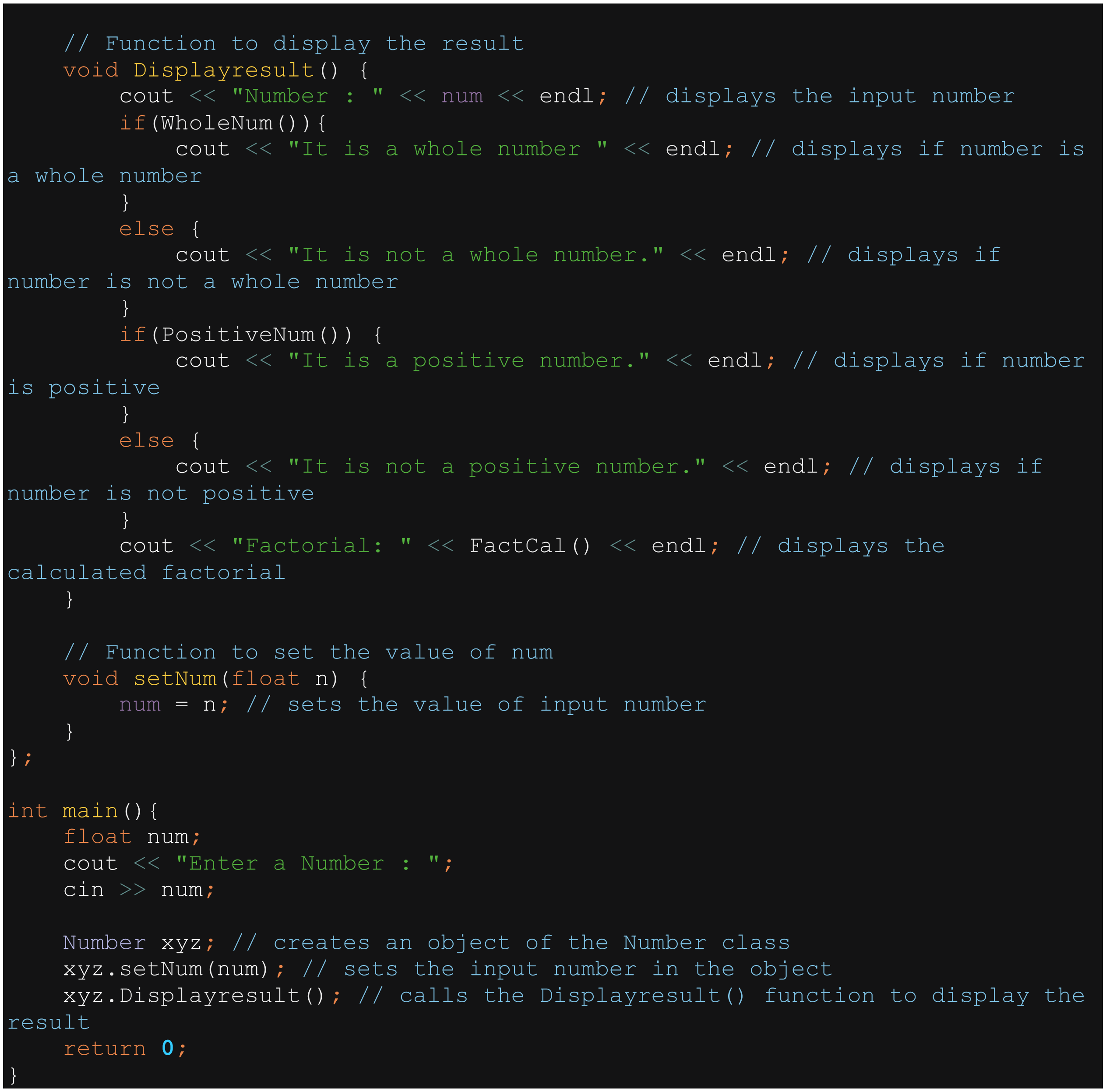
Code no 3:-



```cpp
//
// Created by zohaib on 29/05/2023.
//

/*Task no 1:-->
```

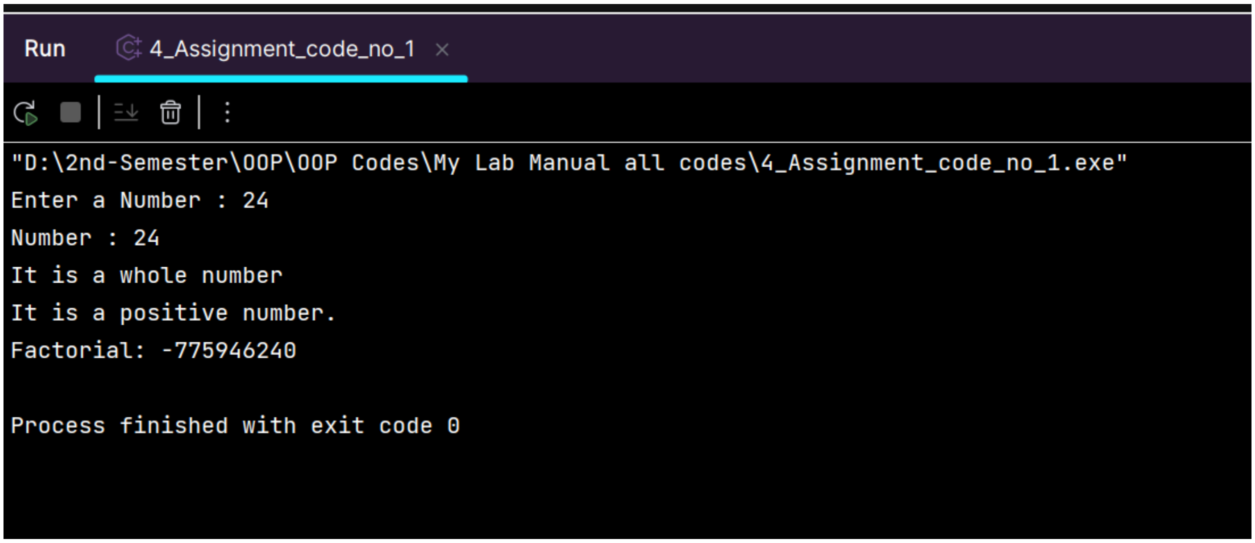


```
Design a program that defines a class named "Number" with two data members,
"num" of type float and
"result" of type int. The program should include four functions that can be
used to calculate the factorial
of a number and display the result:
•   A function to check if the number is a whole number or not.
•   A function to check if the number is positive or not.
•   A function to calculate the factorial of the given number.
•   A function to display the input number and its factorial. Please note
that to calculate
    the factorial of a number, it must be positive and a whole number. If any
of these conditions
    are not met, the program cannot determine the factorial.
*/


#include<iostream>
using namespace std;

class Number {
private:
    float num; // private data member to store the input number
    //  int result;

public:
    // Function to check if number is a whole number
    bool WholeNum() {
        int wholenum = (int)num; // Typecast num to an int to remove decimal
part
        if (num == wholenum){//checks if the original number is equal to the
original value of the number
            return true; // returns true if number is a whole number
        }
        else{
            return false; // returns false if number is not a whole number
        }
    }

    // Function to check if number is positive
    bool PositiveNum() {
        if(num > 0){
            return true; // returns true if number is positive
        }
        else {
            return false; // returns false if number is not positive
        }
    }

    // Function to calculate the factorial of the given number
    int FactCal() {
        int fact = 1;
        for(int i = 1; i <= num; i++) {
            fact *= i; // calculates the factorial of the number
        }
        return fact; // returns the calculated factorial
    }
```



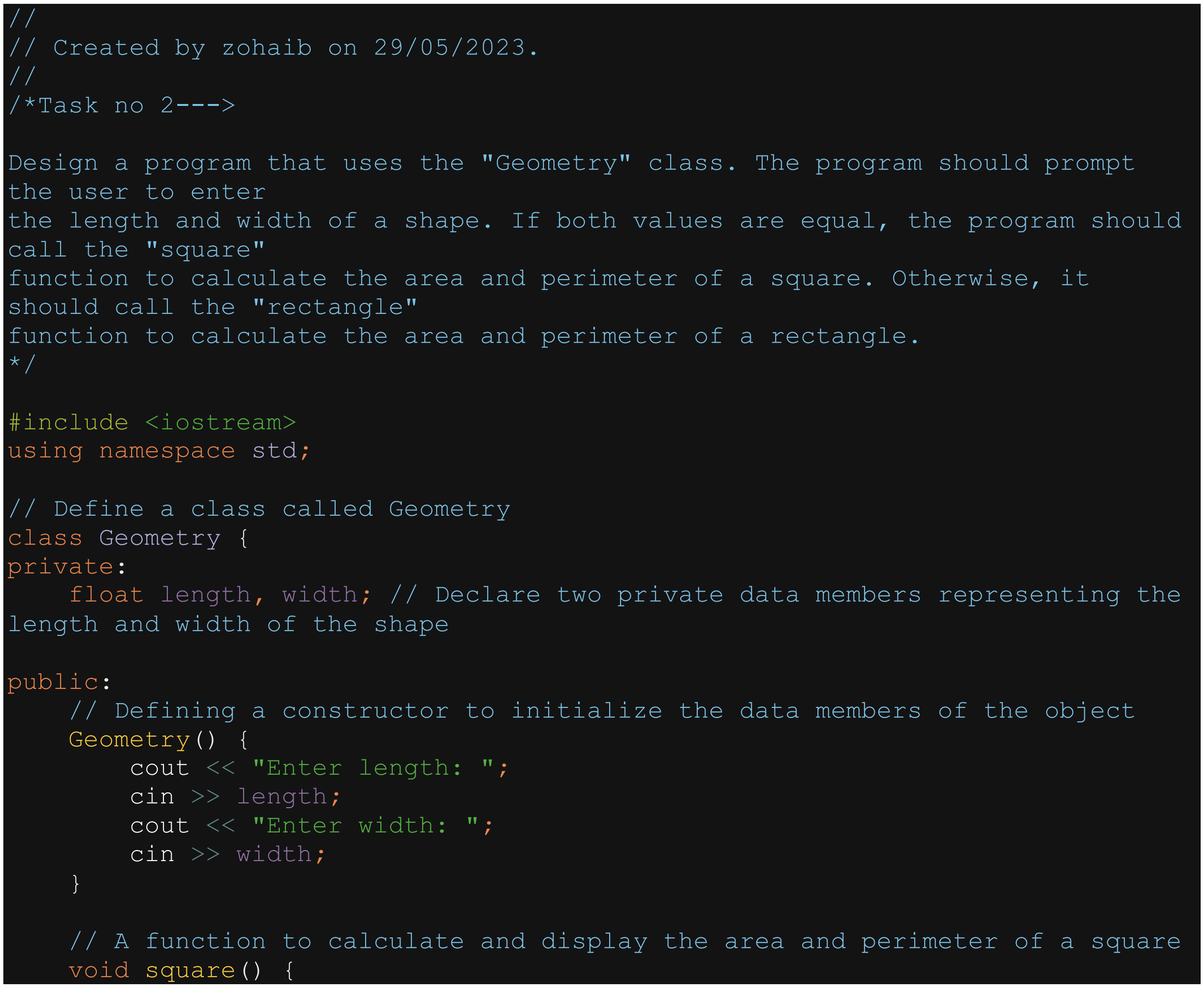
```
    // Function to display the result
    void Displayresult() {
        cout << "Number : " << num << endl; // displays the input number
        if(WholeNum()){
            cout << "It is a whole number " << endl; // displays if number is
a whole number
        }
        else {
            cout << "It is not a whole number." << endl; // displays if
number is not a whole number
        }
        if(PositiveNum()) {
            cout << "It is a positive number." << endl; // displays if number
is positive
        }
        else {
            cout << "It is not a positive number." << endl; // displays if
number is not positive
        }
        cout << "Factorial: " << FactCal() << endl; // displays the
calculated factorial
    }

    // Function to set the value of num
    void setNum(float n) {
        num = n; // sets the value of input number
    }
};

int main(){
    float num;
    cout << "Enter a Number : ";
    cin >> num;

    Number xyz; // creates an object of the Number class
    xyz.setNum(num); // sets the input number in the object
    xyz.Displayresult(); // calls the Displayresult() function to display the
result
    return 0;
}
```

Output:-

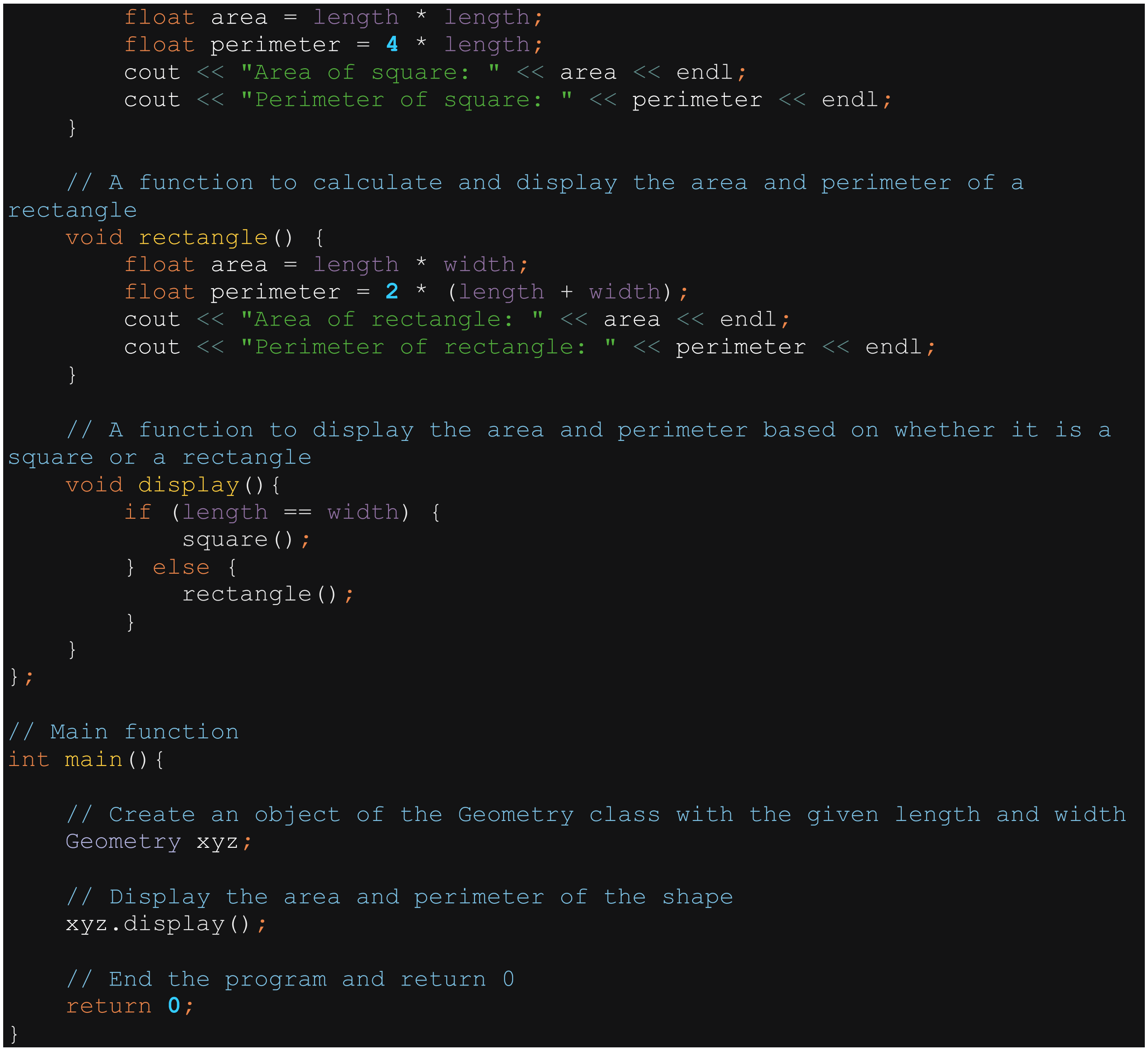


```
"D:\2nd-Semester\OOP\OOP Codes\My Lab Manual all codes\4_Assignment_code_no_1.exe"
Enter a Number : 24
Number : 24
It is a whole number
It is a positive number.
Factorial: -775946240

Process finished with exit code 0
```

Code no 4:-

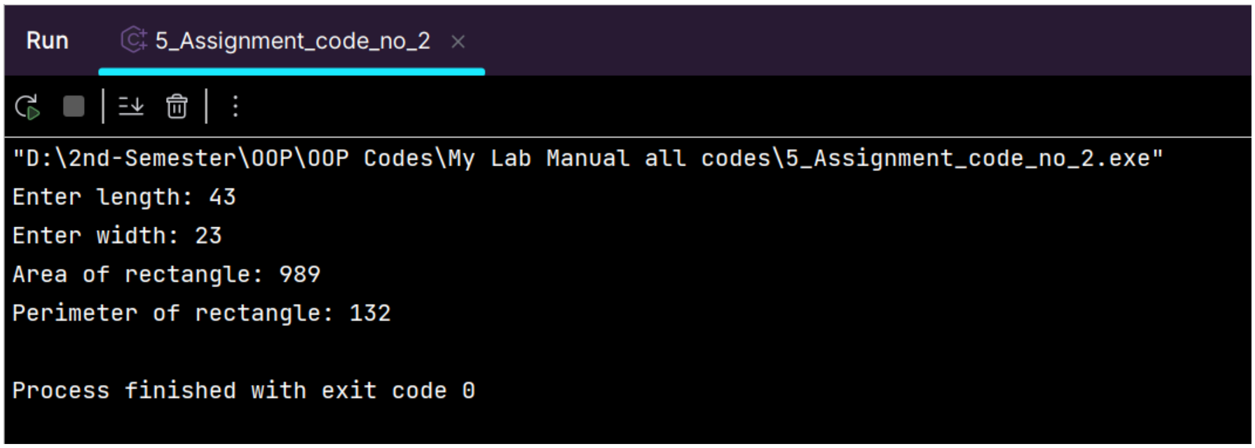


```
//
// Created by zohaib on 29/05/2023.
//
/*Task no 2--->

Design a program that uses the "Geometry" class. The program should prompt
the user to enter
the length and width of a shape. If both values are equal, the program should
call the "square"
function to calculate the area and perimeter of a square. Otherwise, it
should call the "rectangle"
function to calculate the area and perimeter of a rectangle.
*/

#include <iostream>
using namespace std;

// Define a class called Geometry
class Geometry {
private:
    float length, width; // Declare two private data members representing the
length and width of the shape

public:
    // Defining a constructor to initialize the data members of the object
    Geometry() {
        cout << "Enter length: ";
        cin >> length;
        cout << "Enter width: ";
        cin >> width;
    }

    // A function to calculate and display the area and perimeter of a square
    void square() {
```



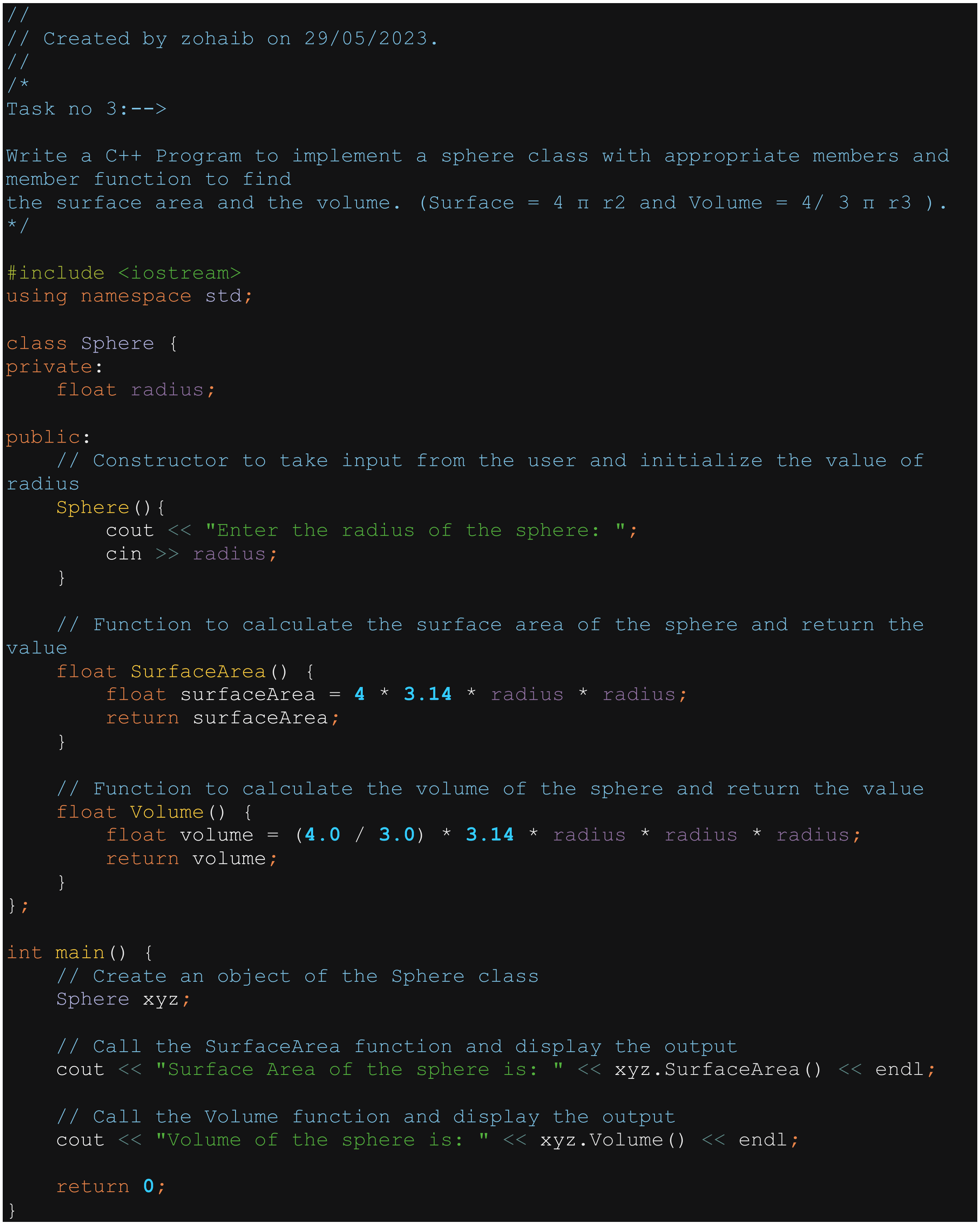
```
        float area = length * length;
        float perimeter = 4 * length;
        cout << "Area of square: " << area << endl;
        cout << "Perimeter of square: " << perimeter << endl;
    }

    // A function to calculate and display the area and perimeter of a
rectangle
    void rectangle() {
        float area = length * width;
        float perimeter = 2 * (length + width);
        cout << "Area of rectangle: " << area << endl;
        cout << "Perimeter of rectangle: " << perimeter << endl;
    }

    // A function to display the area and perimeter based on whether it is a
square or a rectangle
    void display(){
        if (length == width) {
            square();
        } else {
            rectangle();
        }
    }
};

// Main function
int main(){

    // Create an object of the Geometry class with the given length and width
    Geometry xyz;

    // Display the area and perimeter of the shape
    xyz.display();

    // End the program and return 0
    return 0;
}
```

Output:-



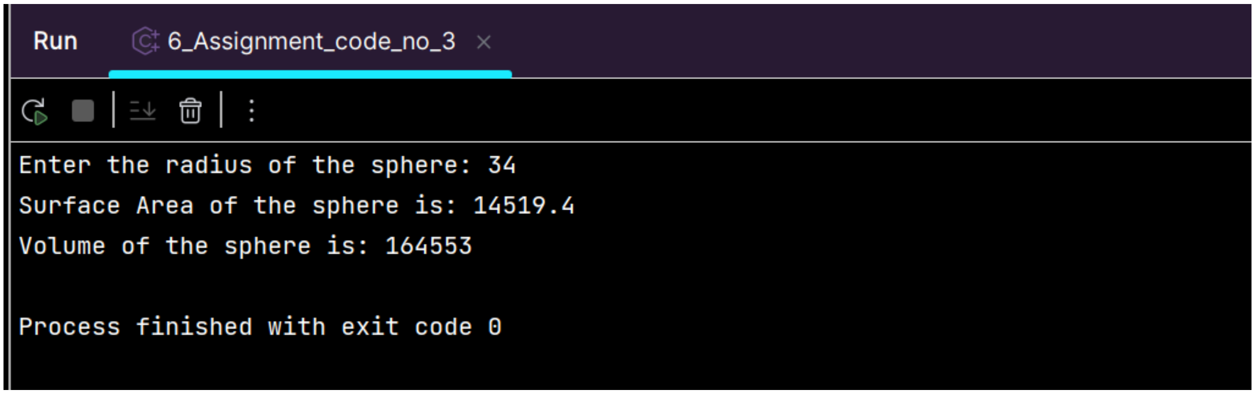
```
"D:\2nd-Semester\OOP\OOP Codes\My Lab Manual all codes\5_Assignment_code_no_2.exe"
Enter length: 43
Enter width: 23
Area of rectangle: 989
Perimeter of rectangle: 132

Process finished with exit code 0
```

Code No 5:-



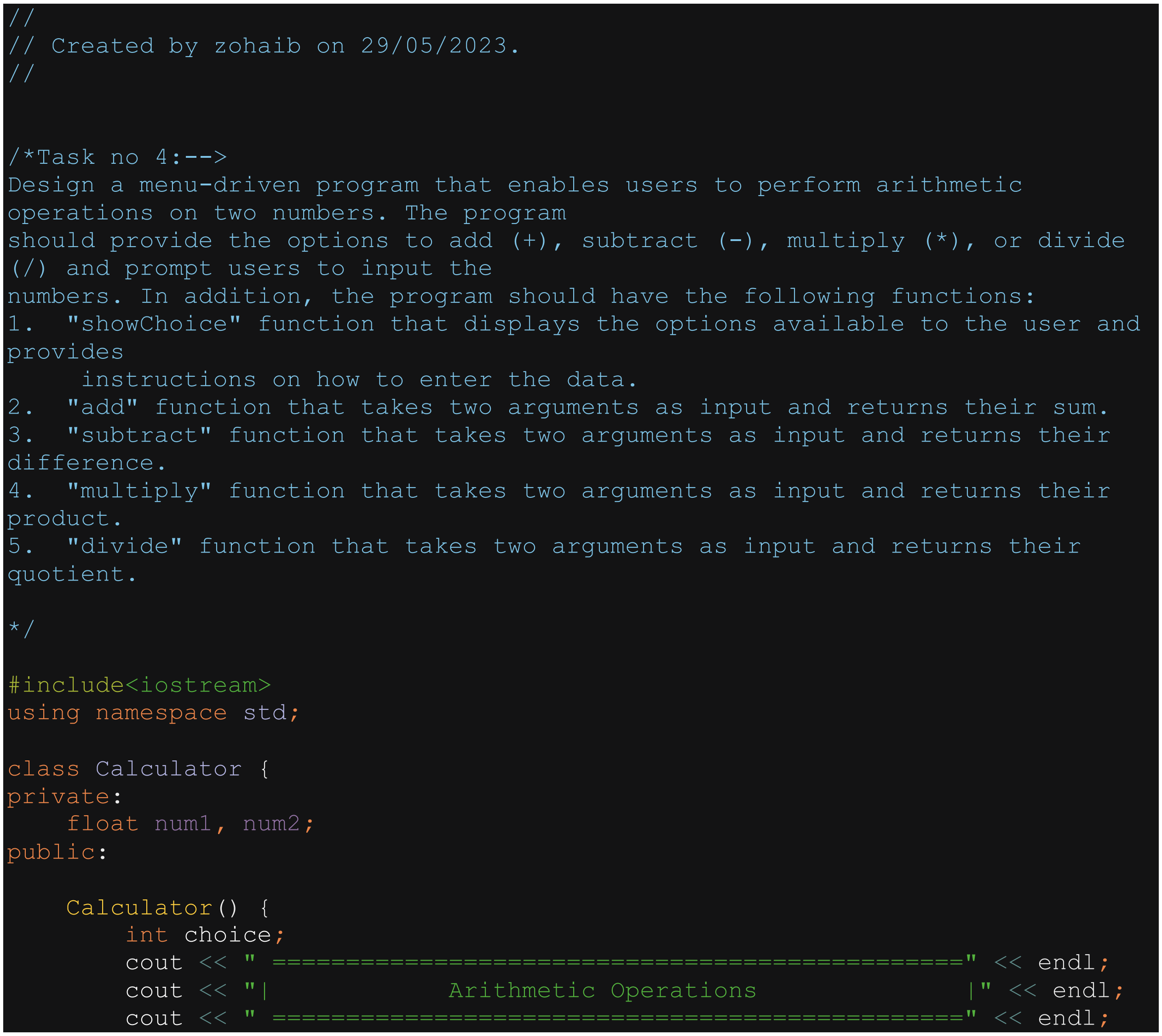
```
//
// Created by zohaib on 29/05/2023.
//
/*
Task no 3:-->

Write a C++ Program to implement a sphere class with appropriate members and
member function to find
the surface area and the volume. (Surface = 4 π r2 and Volume = 4/ 3 π r3 ).
*/

#include <iostream>
using namespace std;

class Sphere {
private:
    float radius;

public:
    // Constructor to take input from the user and initialize the value of
radius
    Sphere(){
        cout << "Enter the radius of the sphere: ";
        cin >> radius;
    }

    // Function to calculate the surface area of the sphere and return the
value
    float SurfaceArea() {
        float surfaceArea = 4 * 3.14 * radius * radius;
        return surfaceArea;
    }

    // Function to calculate the volume of the sphere and return the value
    float Volume() {
        float volume = (4.0 / 3.0) * 3.14 * radius * radius * radius;
        return volume;
    }
};

int main() {
    // Create an object of the Sphere class
    Sphere xyz;

    // Call the SurfaceArea function and display the output
    cout << "Surface Area of the sphere is: " << xyz.SurfaceArea() << endl;

    // Call the Volume function and display the output
    cout << "Volume of the sphere is: " << xyz.Volume() << endl;

    return 0;
}
```

Output:-

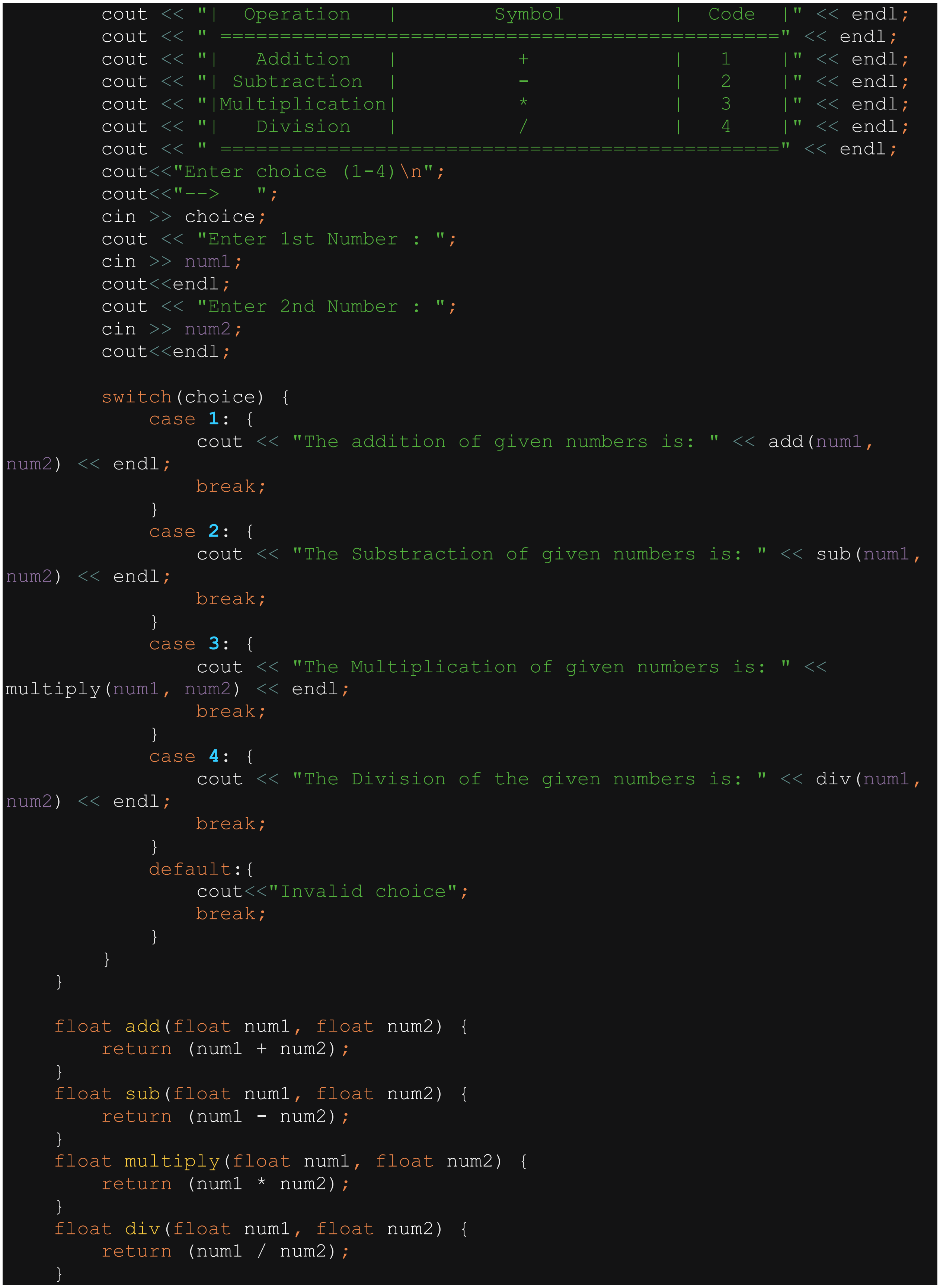


```
Enter the radius of the sphere: 34
Surface Area of the sphere is: 14519.4
Volume of the sphere is: 164553

Process finished with exit code 0
```

Code no 6:-



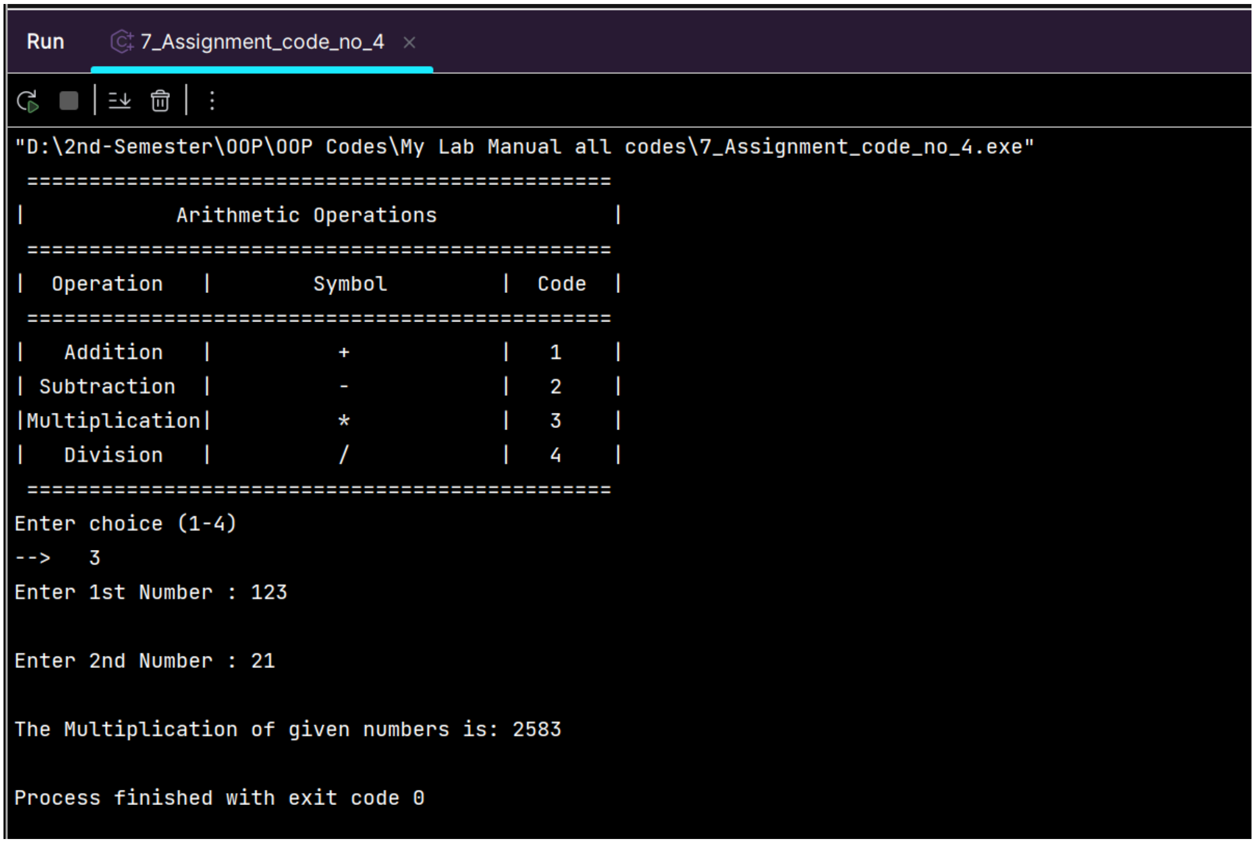
```
//
// Created by zohaib on 29/05/2023.
//


/*Task no 4:-->
Design a menu-driven program that enables users to perform arithmetic
operations on two numbers. The program
should provide the options to add (+), subtract (-), multiply (*), or divide
(/) and prompt users to input the
numbers. In addition, the program should have the following functions:
1.  "showChoice" function that displays the options available to the user and
provides
     instructions on how to enter the data.
2.  "add" function that takes two arguments as input and returns their sum.
3.  "subtract" function that takes two arguments as input and returns their
difference.
4.  "multiply" function that takes two arguments as input and returns their
product.
5.  "divide" function that takes two arguments as input and returns their
quotient.

*/

#include<iostream>
using namespace std;

class Calculator {
private:
    float num1, num2;
public:

    Calculator() {
        int choice;
        cout << " ================================================" << endl;
        cout << "|            Arithmetic Operations               |" << endl;
        cout << " ================================================" << endl;
```



```
        cout << "|  Operation   |        Symbol         |  Code  |" << endl;
        cout << " ===============================================" << endl;
        cout << "|   Addition   |          +            |   1    |" << endl;
        cout << "| Subtraction  |          -            |   2    |" << endl;
        cout << "|Multiplication|          *            |   3    |" << endl;
        cout << "|   Division   |          /            |   4    |" << endl;
        cout << " ===============================================" << endl;
        cout<<"Enter choice (1-4)\n";
        cout<<"-->   ";
        cin >> choice;
        cout << "Enter 1st Number : ";
        cin >> num1;
        cout<<endl;
        cout << "Enter 2nd Number : ";
        cin >> num2;
        cout<<endl;

        switch(choice) {
            case 1: {
                cout << "The addition of given numbers is: " << add(num1,
num2) << endl;
                break;
            }
            case 2: {
                cout << "The Substraction of given numbers is: " << sub(num1,
num2) << endl;
                break;
            }
            case 3: {
                cout << "The Multiplication of given numbers is: " <<
multiply(num1, num2) << endl;
                break;
            }
            case 4: {
                cout << "The Division of the given numbers is: " << div(num1,
num2) << endl;
                break;
            }
            default:{
                cout<<"Invalid choice";
                break;
            }
        }
    }

    float add(float num1, float num2) {
        return (num1 + num2);
    }
    float sub(float num1, float num2) {
        return (num1 - num2);
    }
    float multiply(float num1, float num2) {
        return (num1 * num2);
    }
    float div(float num1, float num2) {
        return (num1 / num2);
    }
```

```
};

int main() {
    Calculator xyz;
    return 0;
}
```

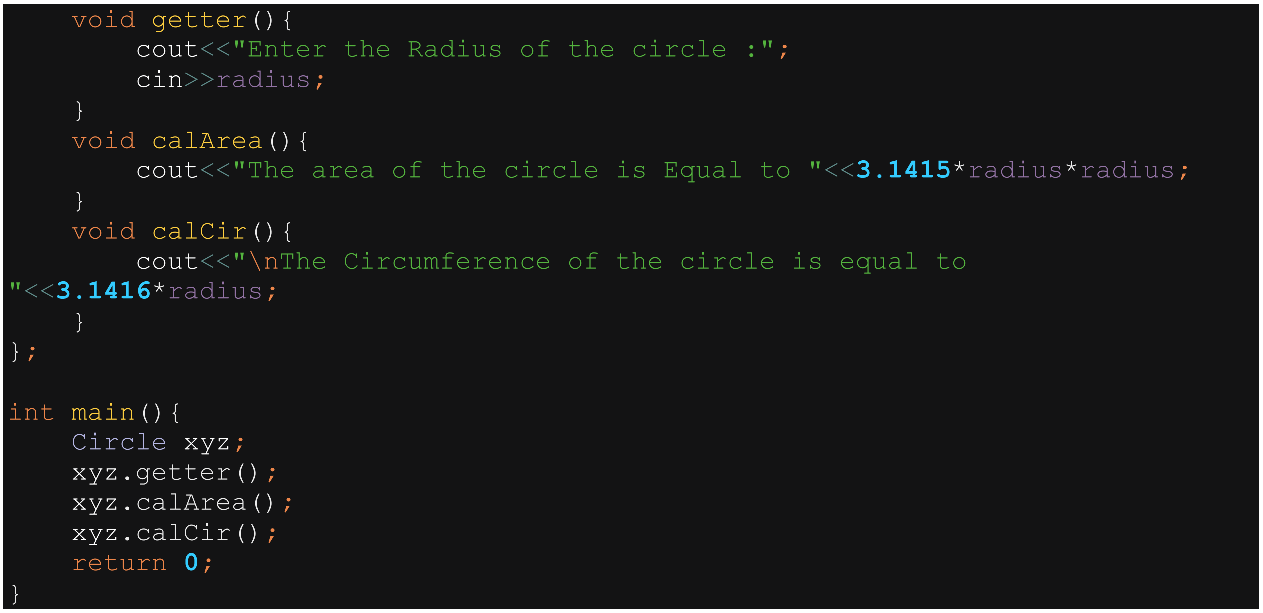
Output:-



```
"D:\2nd-Semester\OOP\OOP Codes\My Lab Manual all codes\7_Assignment_code_no_4.exe"
 ===============================================
|            Arithmetic Operations              |
 ===============================================
|  Operation   |         Symbol         |  Code  |
 ===============================================
|   Addition   |           +            |   1    |
| Subtraction  |           -            |   2    |
|Multiplication|           *            |   3    |
|   Division   |           /            |   4    |
 ===============================================
Enter choice (1-4)
-->   3
Enter 1st Number : 123

Enter 2nd Number : 21

The Multiplication of given numbers is: 2583

Process finished with exit code 0
```

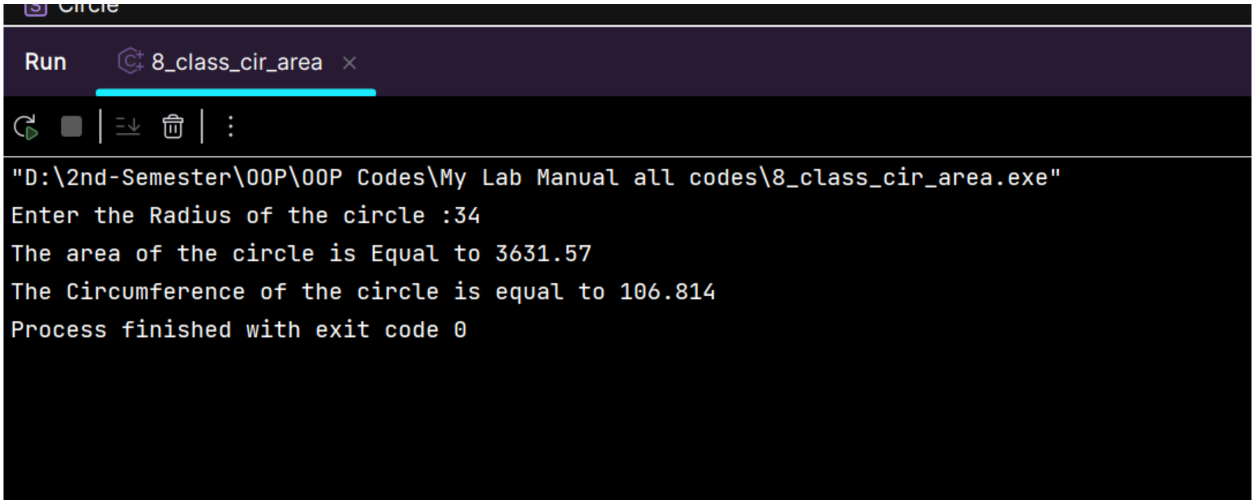
Code no 7:-

```
//
// Created by zohaib on 29/05/2023.
//
/*Q#.7:Create a Class named as Circle. Create a data member named as radius.
Create member functions
to calculate area of circle and circumference of the circle.*/

#include <iostream>
using namespace std;

class Circle{
private:
    float radius;
public:
```



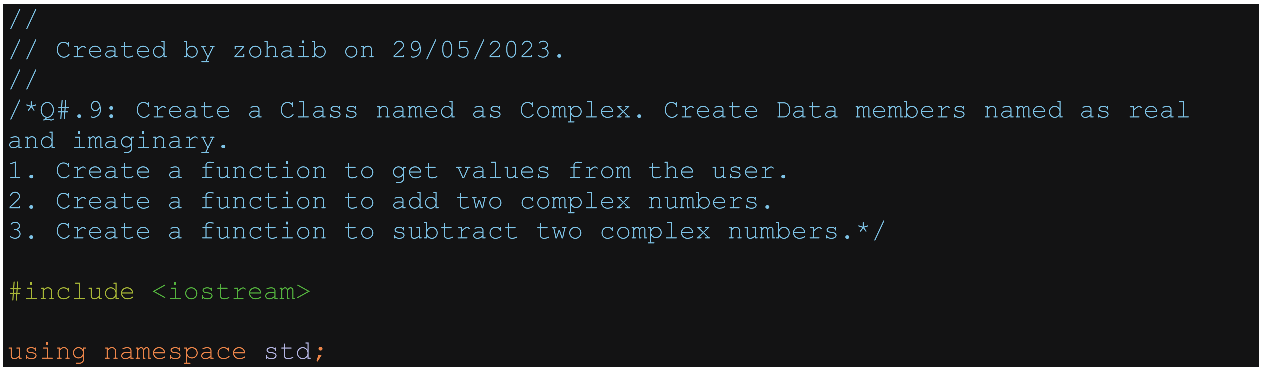
```
    void getter(){
        cout<<"Enter the Radius of the circle :";
        cin>>radius;
    }
    void calArea(){
        cout<<"The area of the circle is Equal to "<<3.1415*radius*radius;
    }
    void calCir(){
        cout<<"\nThe Circumference of the circle is equal to
"<<3.1416*radius;
    }
};

int main(){
    Circle xyz;
    xyz.getter();
    xyz.calArea();
    xyz.calCir();
    return 0;
}
```

Output:-



```
"D:\2nd-Semester\OOP\OOP Codes\My Lab Manual all codes\8_class_cir_area.exe"
Enter the Radius of the circle :34
The area of the circle is Equal to 3631.57
The Circumference of the circle is equal to 106.814
Process finished with exit code 0
```

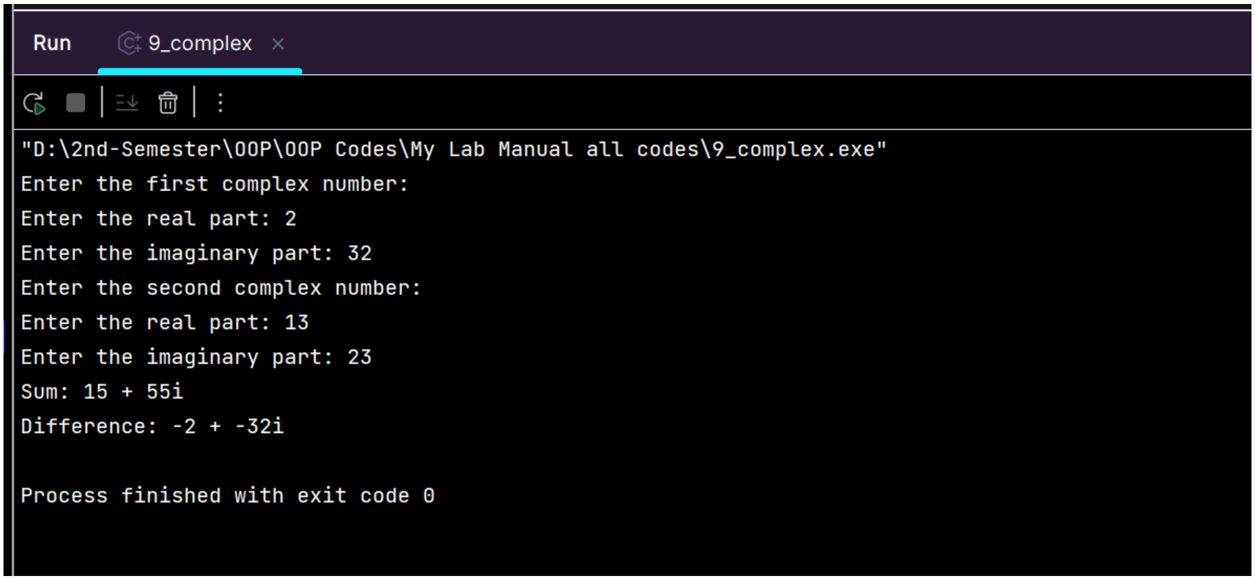
Code no 8:-



```
//
// Created by zohaib on 29/05/2023.
//
/*Q#.9: Create a Class named as Complex. Create Data members named as real
and imaginary.
1. Create a function to get values from the user.
2. Create a function to add two complex numbers.
3. Create a function to subtract two complex numbers.*/

#include <iostream>

using namespace std;
```

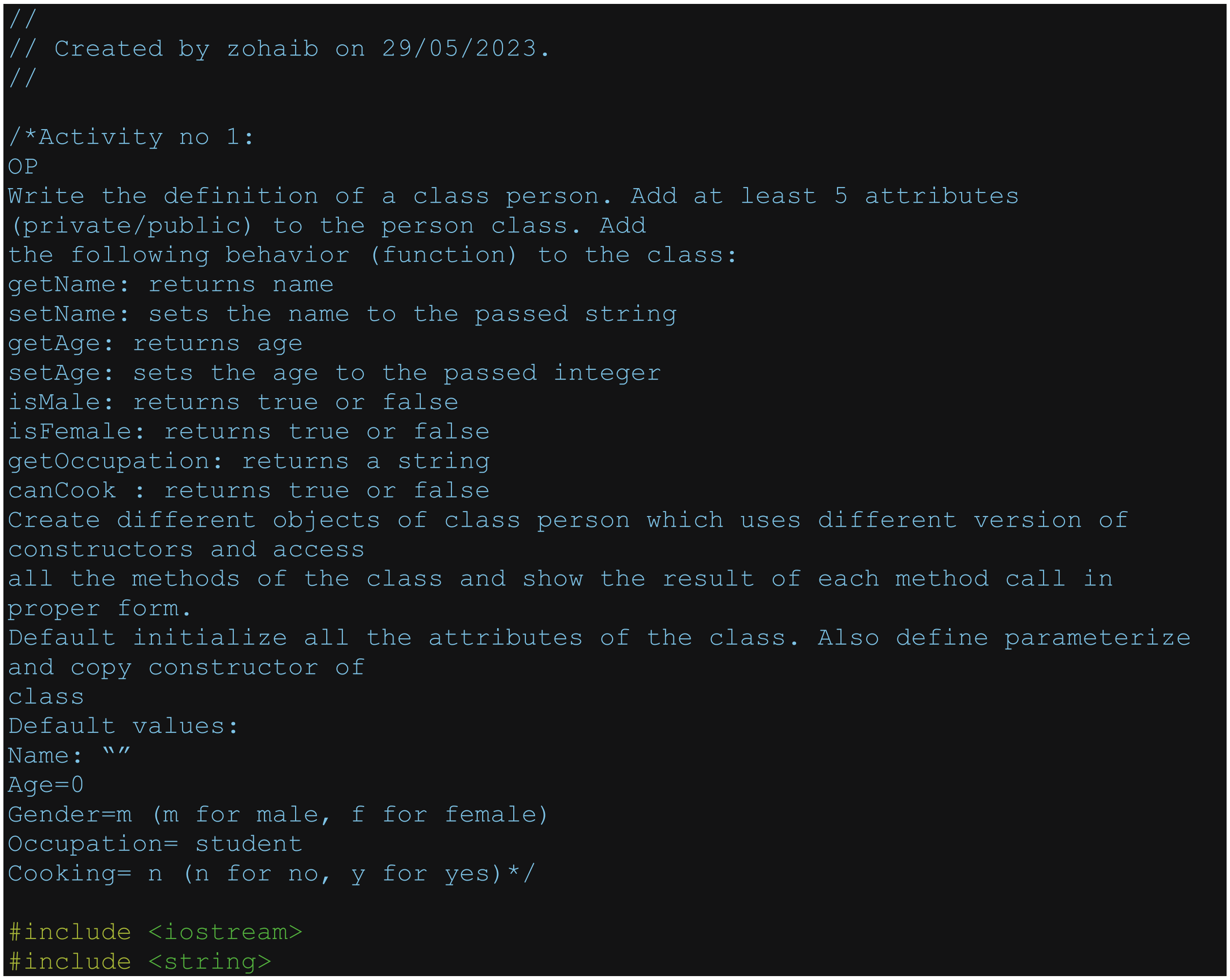
```
class Complex {
private:
    double real;
    double imaginary;

public:
    void getValues() {
        cout << "Enter the real part: ";
        cin >> real;
        cout << "Enter the imaginary part: ";
        cin >> imaginary;
    }

    void add(Complex other) {
        real += other.real;
        imaginary += other.imaginary;
    }

    void subtract(Complex other) {
        real -= other.real;
        imaginary -= other.imaginary;
    }

    double getReal() const {
        return real;
    }

    double getImaginary() const {
        return imaginary;
    }
};

int main() {
    Complex xyz1, xyz2;
    cout << "Enter the first complex number:\n";
    xyz1.getValues();
    cout << "Enter the second complex number:\n";
    xyz2.getValues();

    xyz1.add(xyz2);
    xyz2.subtract(xyz1);

    cout << "Sum: " << xyz1.getReal() << " + " << xyz1.getImaginary() <<
"i\n";
    cout << "Difference: " << xyz2.getReal() << " + " << xyz2.getImaginary()
<< "i\n";

    return 0;
}
```

Output:-

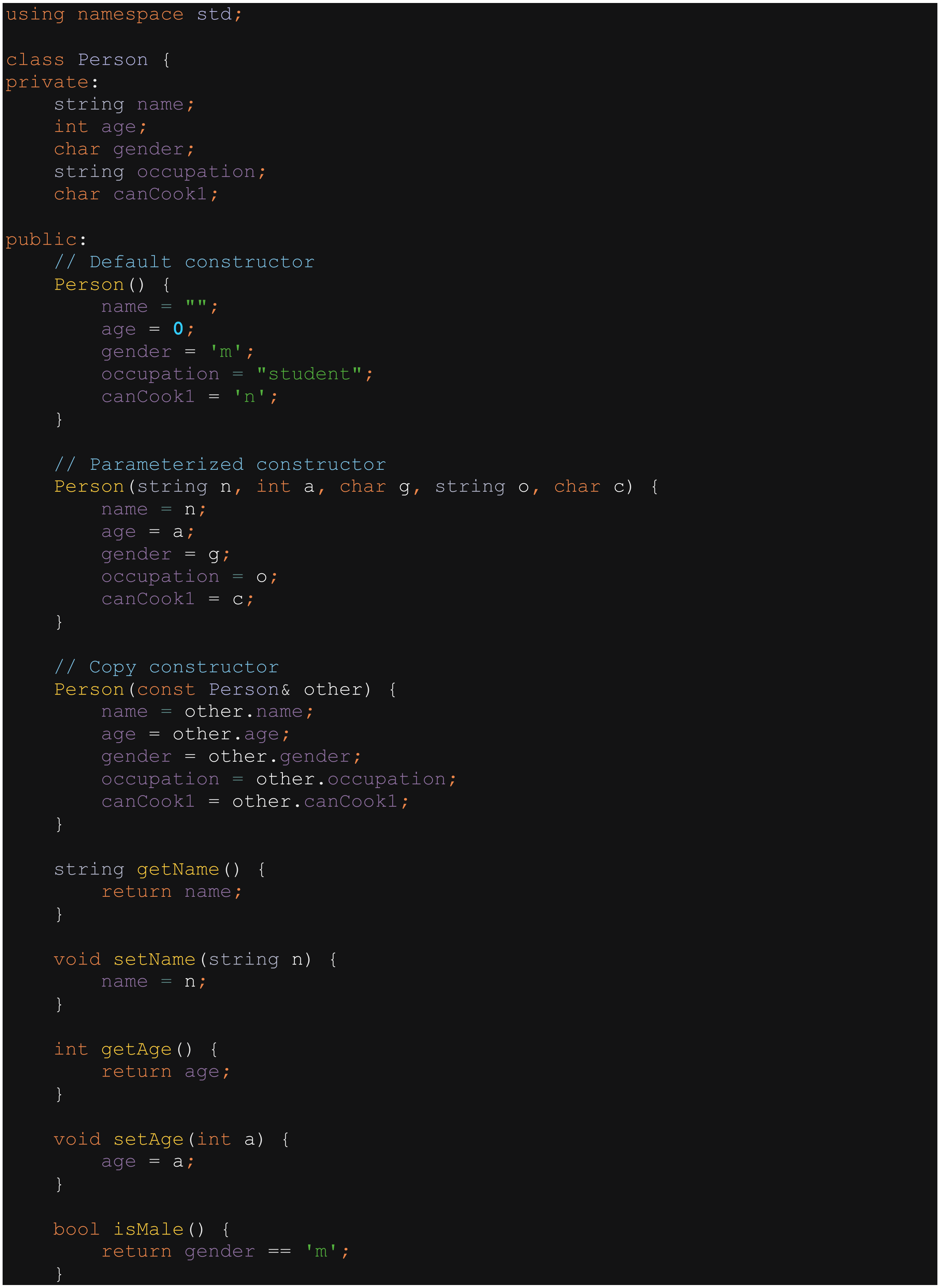


```
"D:\2nd-Semester\OOP\OOP Codes\My Lab Manual all codes\9_complex.exe"
Enter the first complex number:
Enter the real part: 2
Enter the imaginary part: 32
Enter the second complex number:
Enter the real part: 13
Enter the imaginary part: 23
Sum: 15 + 55i
Difference: -2 + -32i

Process finished with exit code 0
```

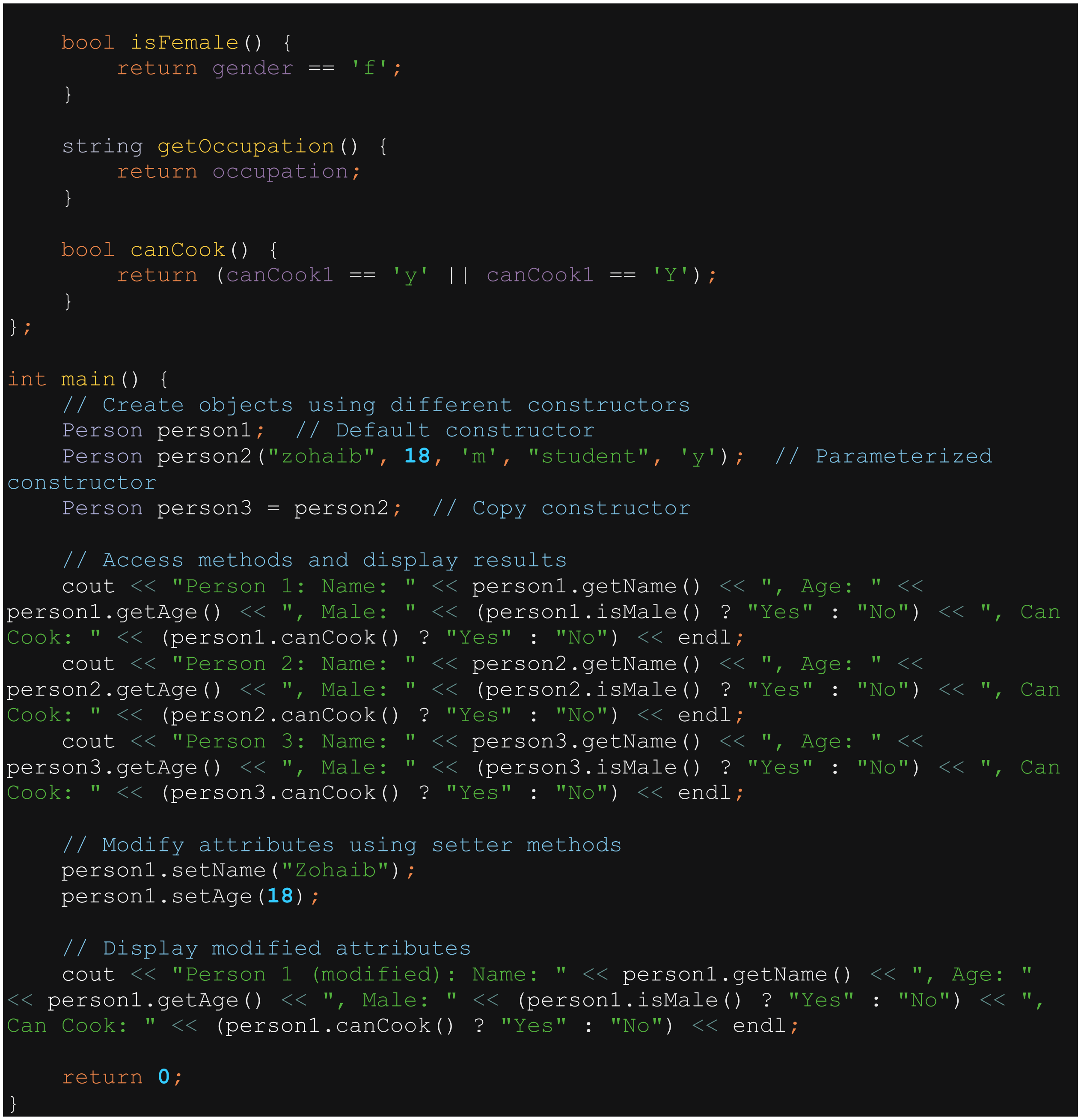
Code no 9:-



```
//
// Created by zohaib on 29/05/2023.
//

/*Activity no 1:
OP
Write the definition of a class person. Add at least 5 attributes
(private/public) to the person class. Add
the following behavior (function) to the class:
getName: returns name
setName: sets the name to the passed string
getAge: returns age
setAge: sets the age to the passed integer
isMale: returns true or false
isFemale: returns true or false
getOccupation: returns a string
canCook : returns true or false
Create different objects of class person which uses different version of
constructors and access
all the methods of the class and show the result of each method call in
proper form.
Default initialize all the attributes of the class. Also define parameterize
and copy constructor of
class
Default values:
Name: “”
Age=0
Gender=m (m for male, f for female)
Occupation= student
Cooking= n (n for no, y for yes)*/

#include <iostream>
#include <string>
```

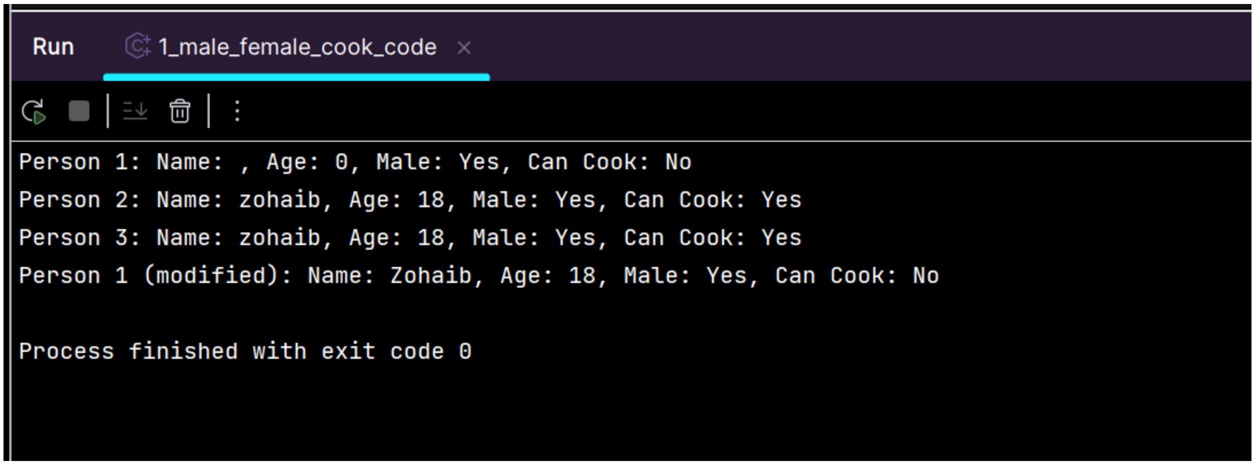


```cpp
using namespace std;

class Person {
private:
    string name;
    int age;
    char gender;
    string occupation;
    char canCook1;

public:
    // Default constructor
    Person() {
        name = "";
        age = 0;
        gender = 'm';
        occupation = "student";
        canCook1 = 'n';
    }

    // Parameterized constructor
    Person(string n, int a, char g, string o, char c) {
        name = n;
        age = a;
        gender = g;
        occupation = o;
        canCook1 = c;
    }

    // Copy constructor
    Person(const Person& other) {
        name = other.name;
        age = other.age;
        gender = other.gender;
        occupation = other.occupation;
        canCook1 = other.canCook1;
    }

    string getName() {
        return name;
    }

    void setName(string n) {
        name = n;
    }

    int getAge() {
        return age;
    }

    void setAge(int a) {
        age = a;
    }

    bool isMale() {
        return gender == 'm';
    }
```



```
    bool isFemale() {
        return gender == 'f';
    }

    string getOccupation() {
        return occupation;
    }

    bool canCook() {
        return (canCook1 == 'y' || canCook1 == 'Y');
    }
};

int main() {
    // Create objects using different constructors
    Person person1;  // Default constructor
    Person person2("zohaib", 18, 'm', "student", 'y');  // Parameterized
constructor
    Person person3 = person2;  // Copy constructor

    // Access methods and display results
    cout << "Person 1: Name: " << person1.getName() << ", Age: " <<
person1.getAge() << ", Male: " << (person1.isMale() ? "Yes" : "No") << ", Can
Cook: " << (person1.canCook() ? "Yes" : "No") << endl;
    cout << "Person 2: Name: " << person2.getName() << ", Age: " <<
person2.getAge() << ", Male: " << (person2.isMale() ? "Yes" : "No") << ", Can
Cook: " << (person2.canCook() ? "Yes" : "No") << endl;
    cout << "Person 3: Name: " << person3.getName() << ", Age: " <<
person3.getAge() << ", Male: " << (person3.isMale() ? "Yes" : "No") << ", Can
Cook: " << (person3.canCook() ? "Yes" : "No") << endl;

    // Modify attributes using setter methods
    person1.setName("Zohaib");
    person1.setAge(18);

    // Display modified attributes
    cout << "Person 1 (modified): Name: " << person1.getName() << ", Age: "
<< person1.getAge() << ", Male: " << (person1.isMale() ? "Yes" : "No") << ",
Can Cook: " << (person1.canCook() ? "Yes" : "No") << endl;

    return 0;
}
```

Output:-



```
Person 1: Name: , Age: 0, Male: Yes, Can Cook: No
Person 2: Name: zohaib, Age: 18, Male: Yes, Can Cook: Yes
Person 3: Name: zohaib, Age: 18, Male: Yes, Can Cook: Yes
Person 1 (modified): Name: Zohaib, Age: 18, Male: Yes, Can Cook: No

Process finished with exit code 0
```