### Class Constructor

1. \*Default Constructor:\*

Write a class Student with a default constructor that initializes the student's name to "Unknown" and age to 0. Add a method display to print the student's details.

#include <iostream>

#include <string>

class Student {

public:

Student() : name("Unknown"), age(0) {}

void display() const {

std::cout << "Name: " << name << ", Age: " << age << std::endl;

}

private:

std::string name;

int age;

};

int main() {

Student student;

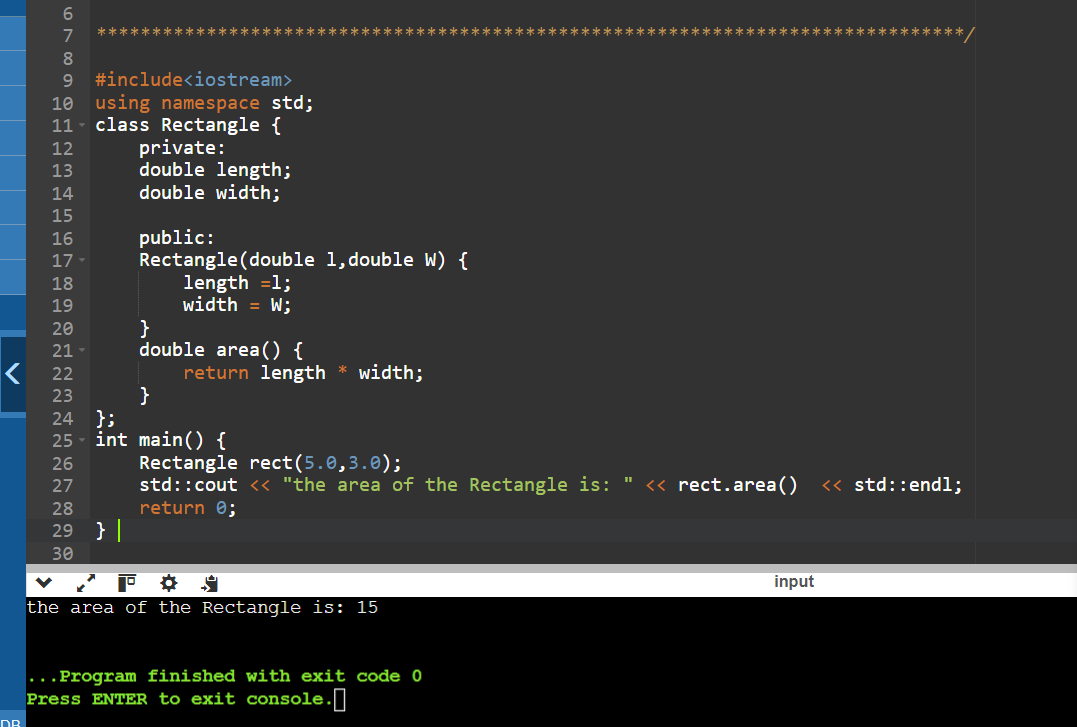
student.display();

return 0;

}

2. \*Parameterized Constructor:\*

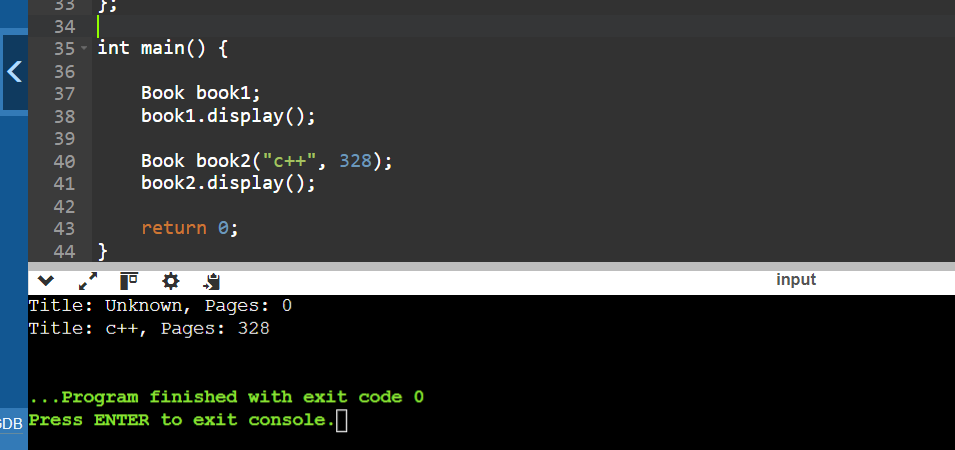
Write a class Rectangle with a parameterized constructor that initializes the length and width. Add a method area that returns the area of the rectangle.



3. \*Multiple Constructors:\*

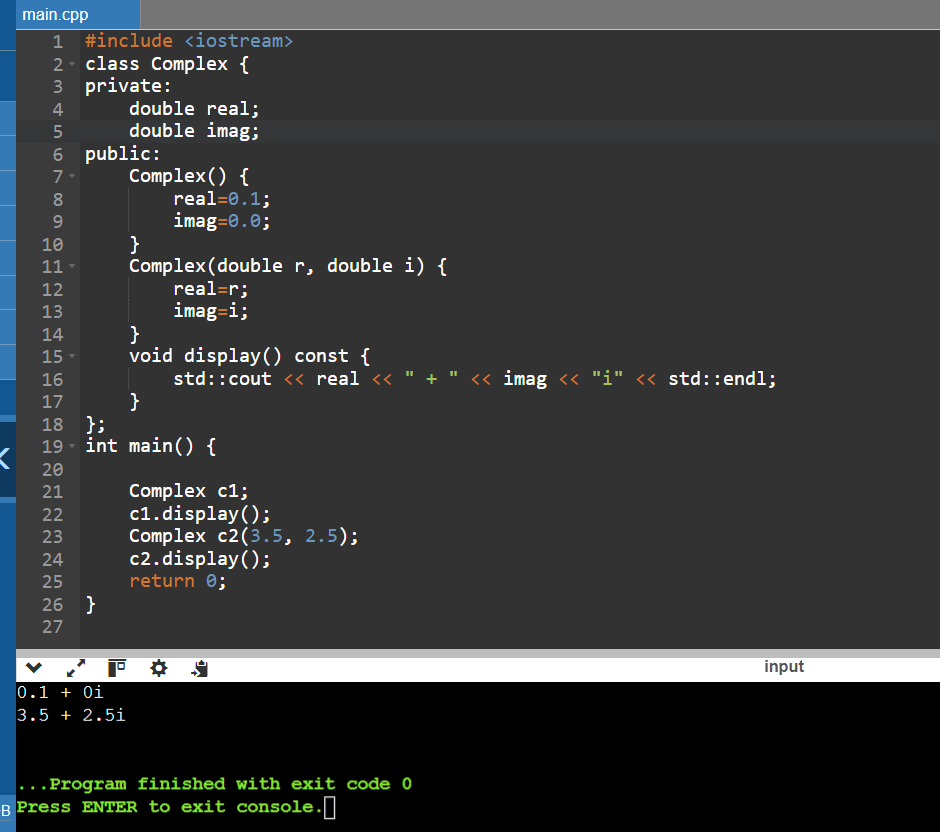
Write a class Book that has both a default constructor and a parameterized constructor. The default constructor should set the title to "Unknown" and the number of pages to 0. The parameterized constructor should initialize the title and pages with given values.





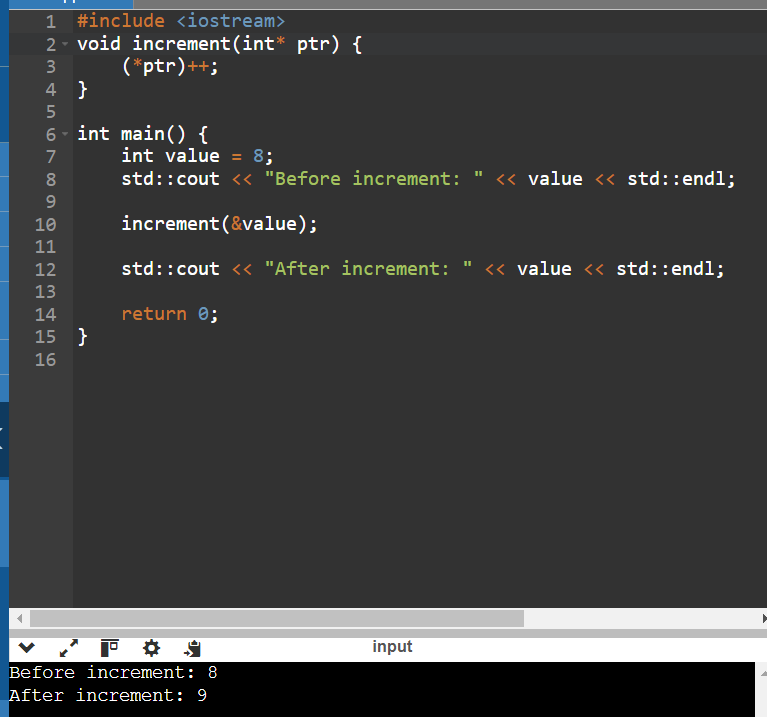
4. \*Constructor Overloading:\*

Write a class Complex that represents complex numbers. Implement a default constructor that sets both real and imaginary parts to 0, and a parameterized constructor that takes two arguments to initialize the real and imaginary parts.



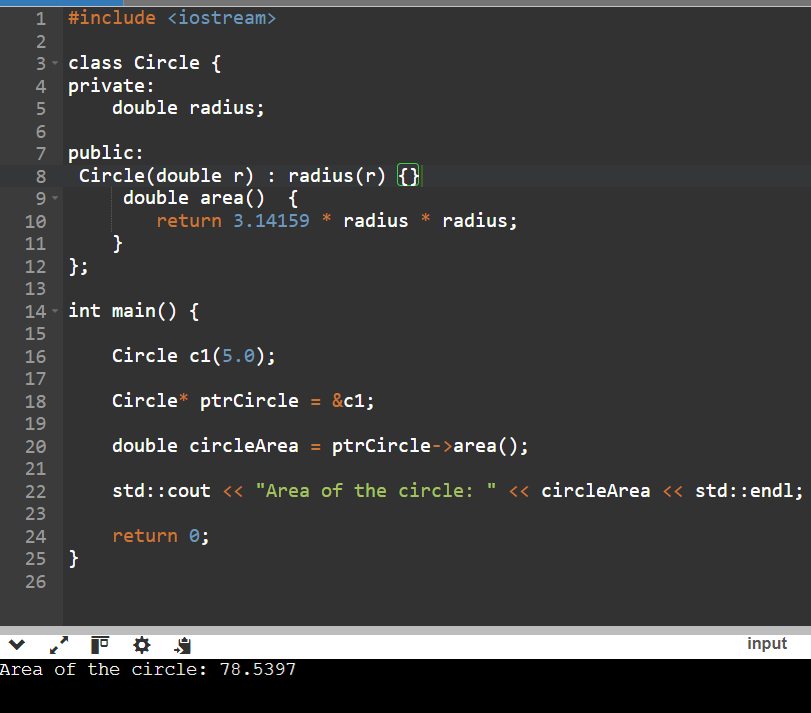
9. \*Pointer to an Integer:\*

Write a function increment that takes a pointer to an integer and increments its value by 1. Demonstrate the function in the main program.



10. \*Pointer to a Class:\*

Write a class Circle with a method area. Create a pointer to an object of this class and call the area method using the pointer.



11. \*Array of Pointers:\*

Write a program that creates an array of pointers to integers. Initialize the array with values and print them using the pointers

