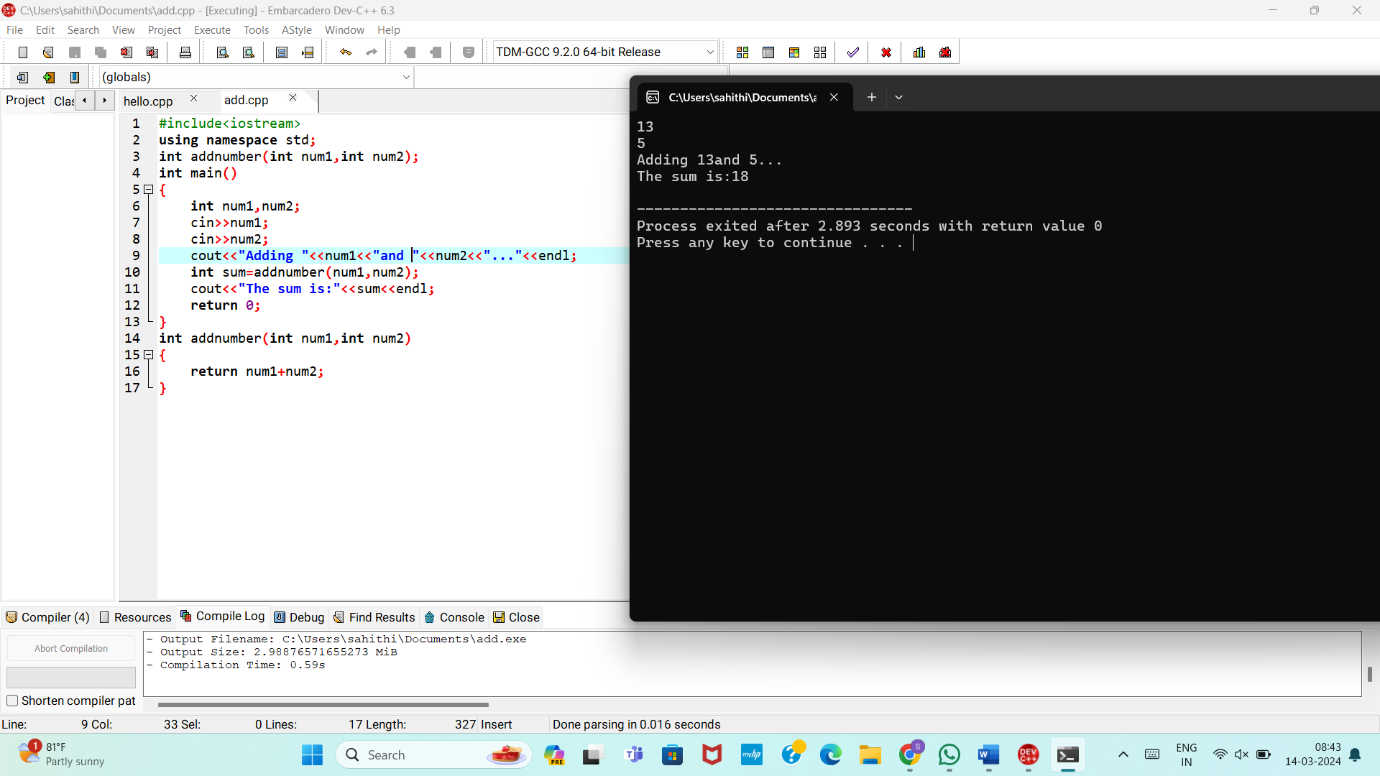
DATE:14-03-2024

EASY PROGRAMS:

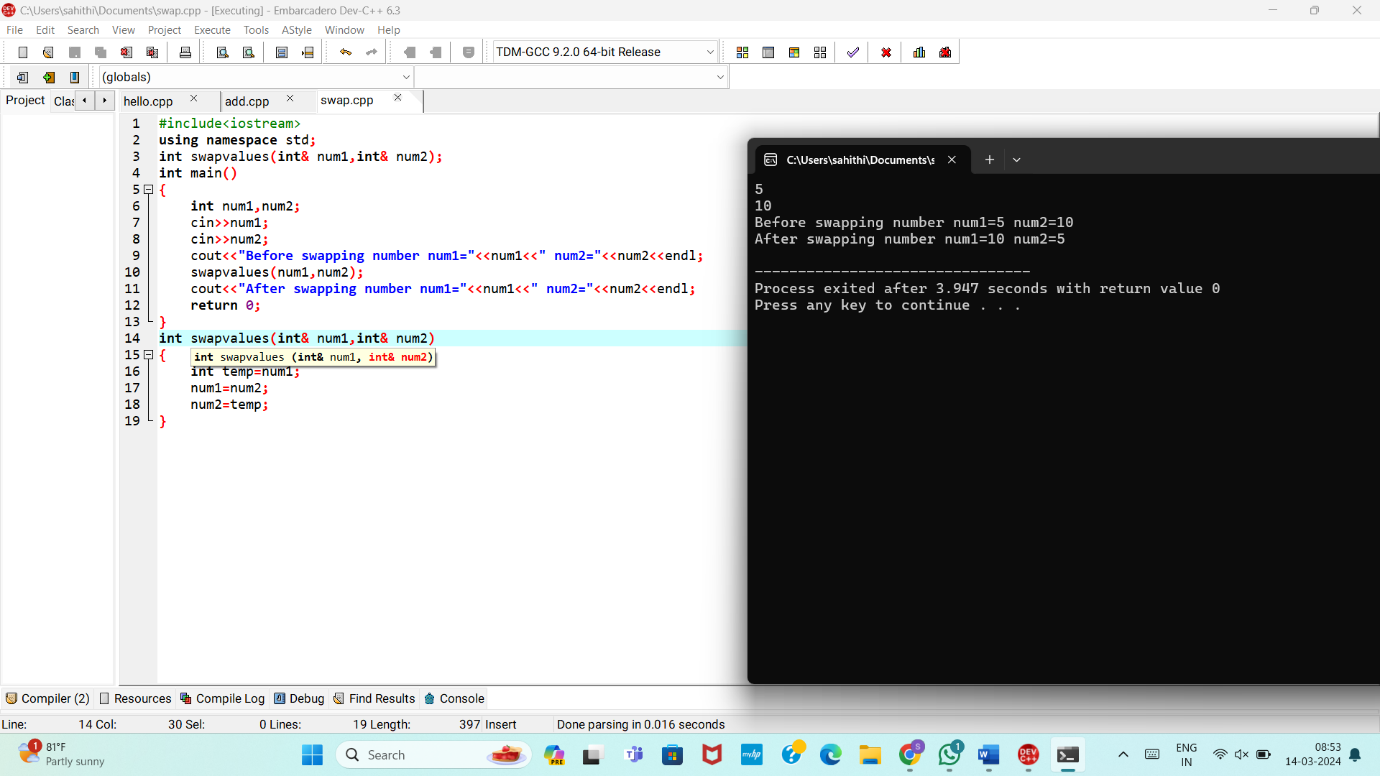
1.Write a C++ program with a main function that prints "Hello, World!" to the console.



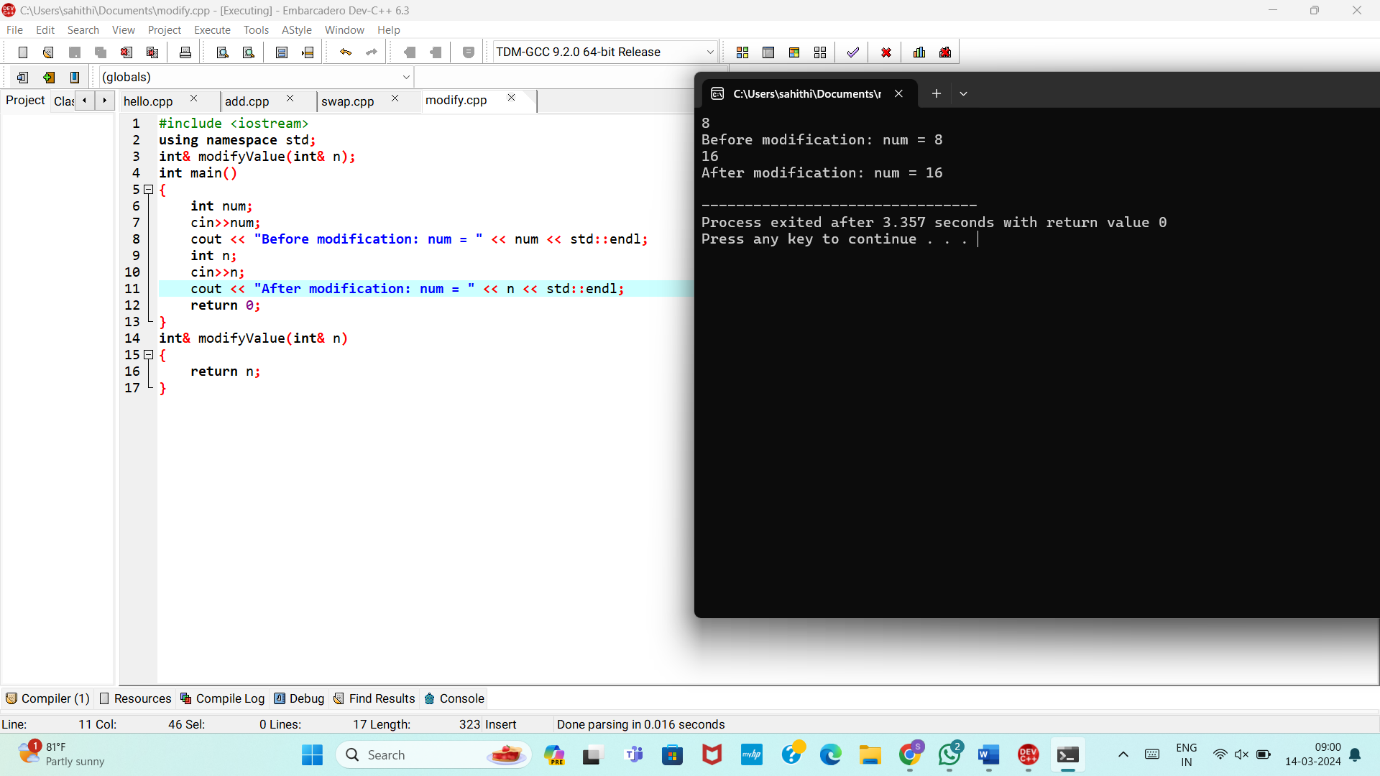
2. Create a C++ program with a function prototype for a function named addNumbers that takes two integers as parameters and returns their sum. Implement the function below the main function and use it to add two numbers.



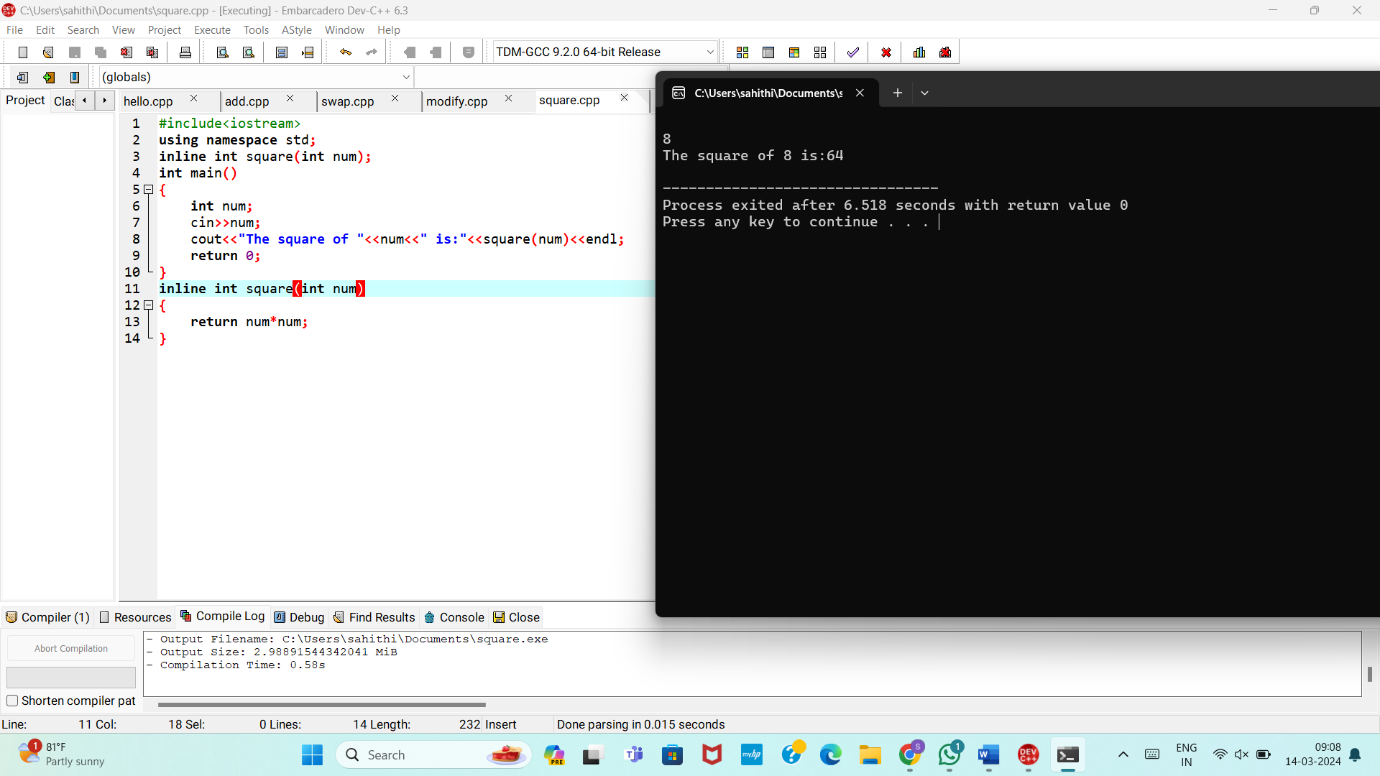
3. Write a C++ program that demonstrates call by reference by swapping the values of two variables.



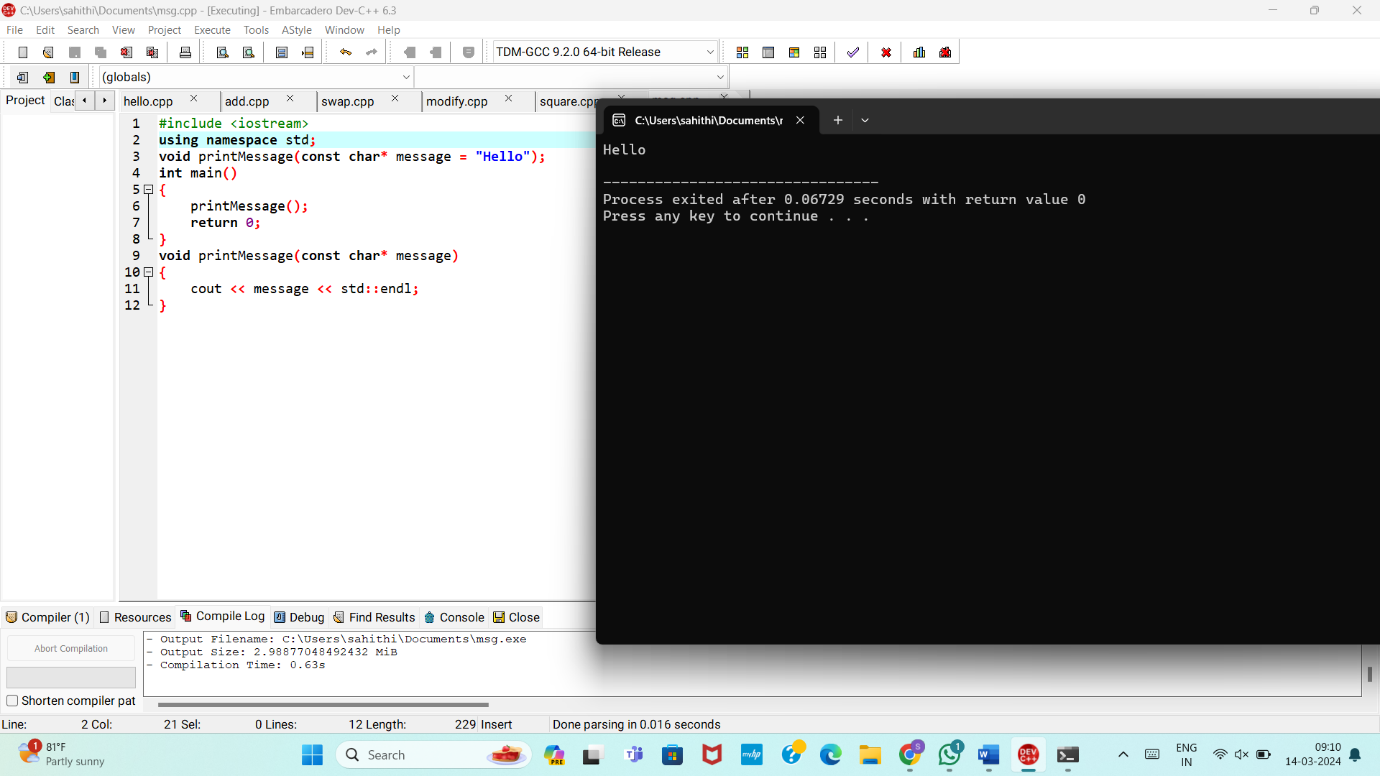
4. Develop a C++ program with a function that returns a reference to an integer variable and modifies its value. Use this function to update the value of a variable in the main function.



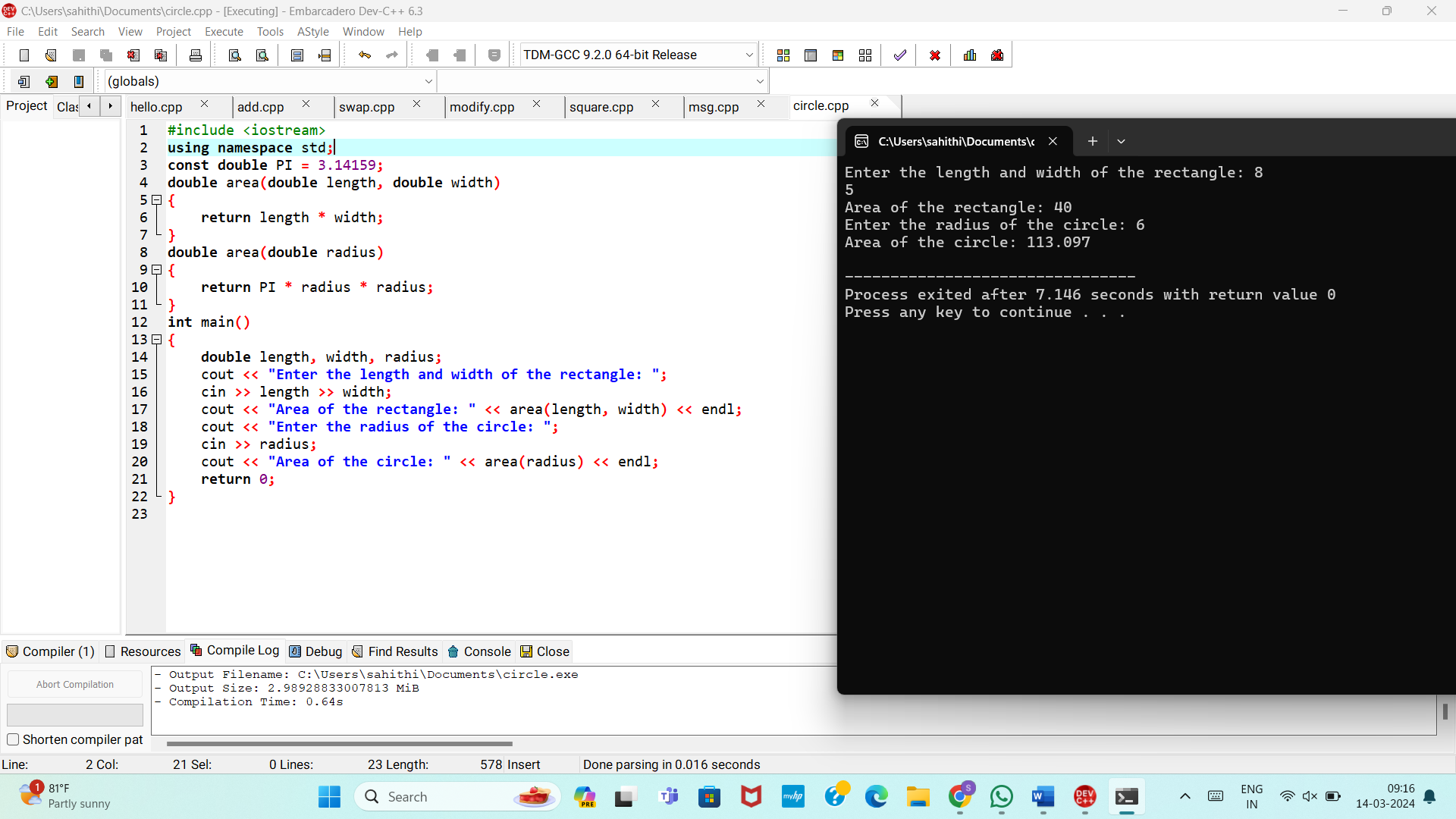
5. Implement an inline function named square that calculates the square of a number. Use this function to square a user-input integer.



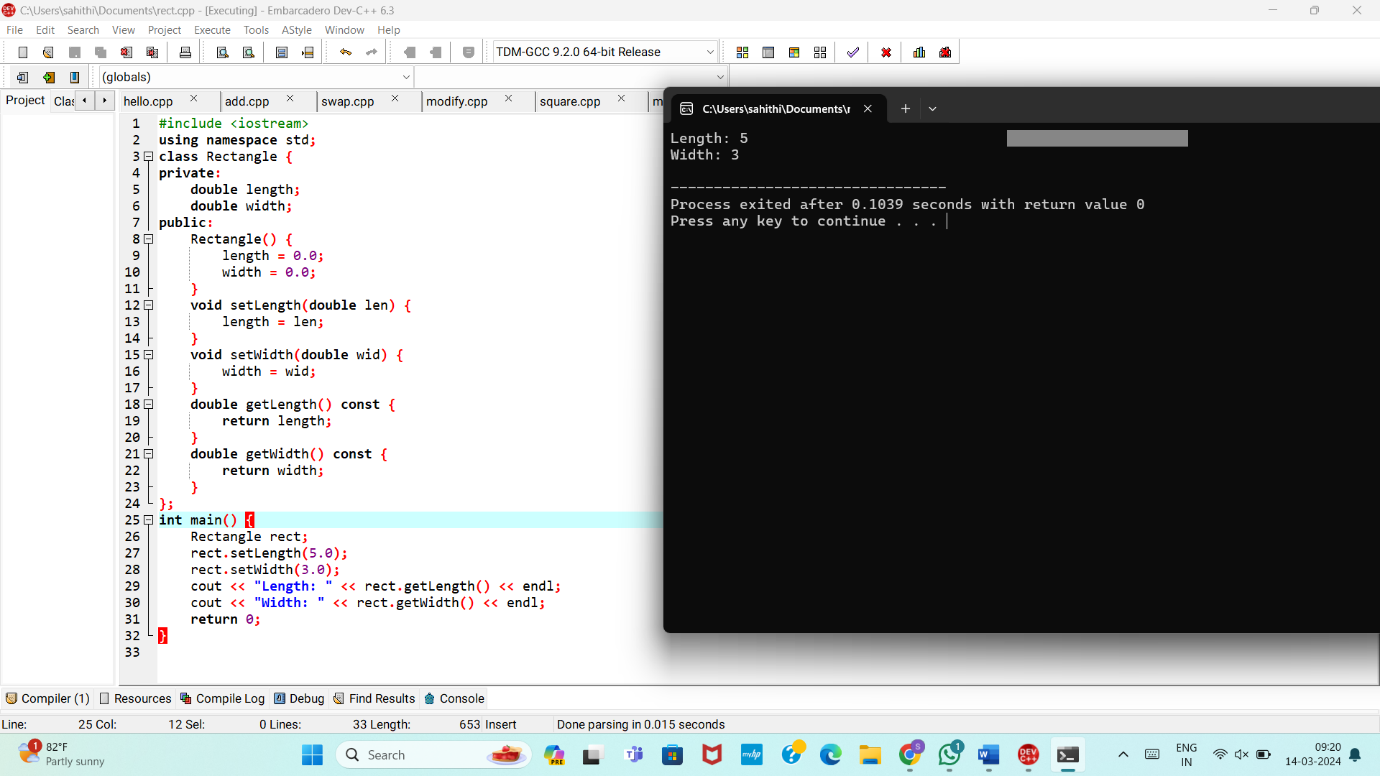
6. Write a C++ program that defines a function named printMessage with a default argument "Hello". Call this function without passing any argument.



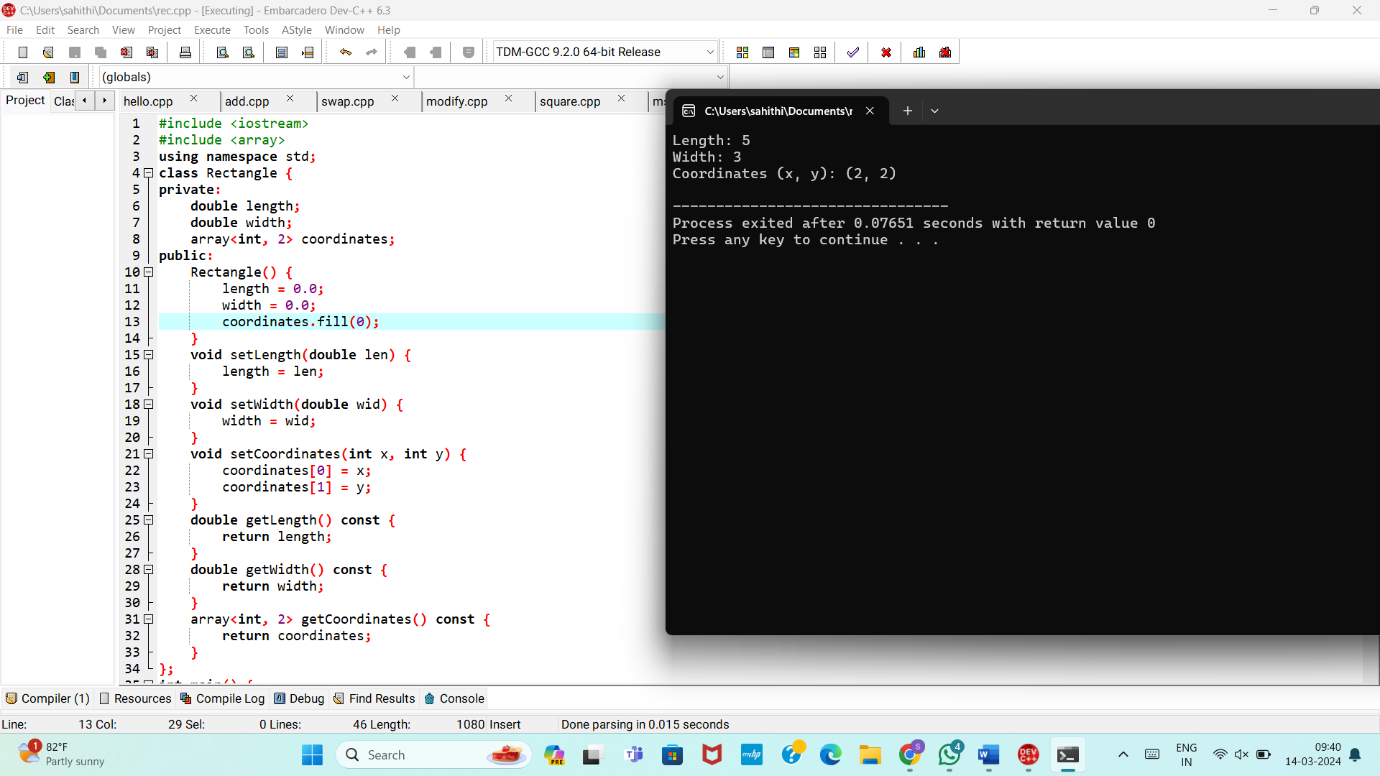
7. Create a C++ program with two overloaded functions named area - one to calculate the area of a rectangle (length \* width) and another to calculate the area of a circle (π \* radius^2). Use function overloading to determine which function to call based on the number of arguments.



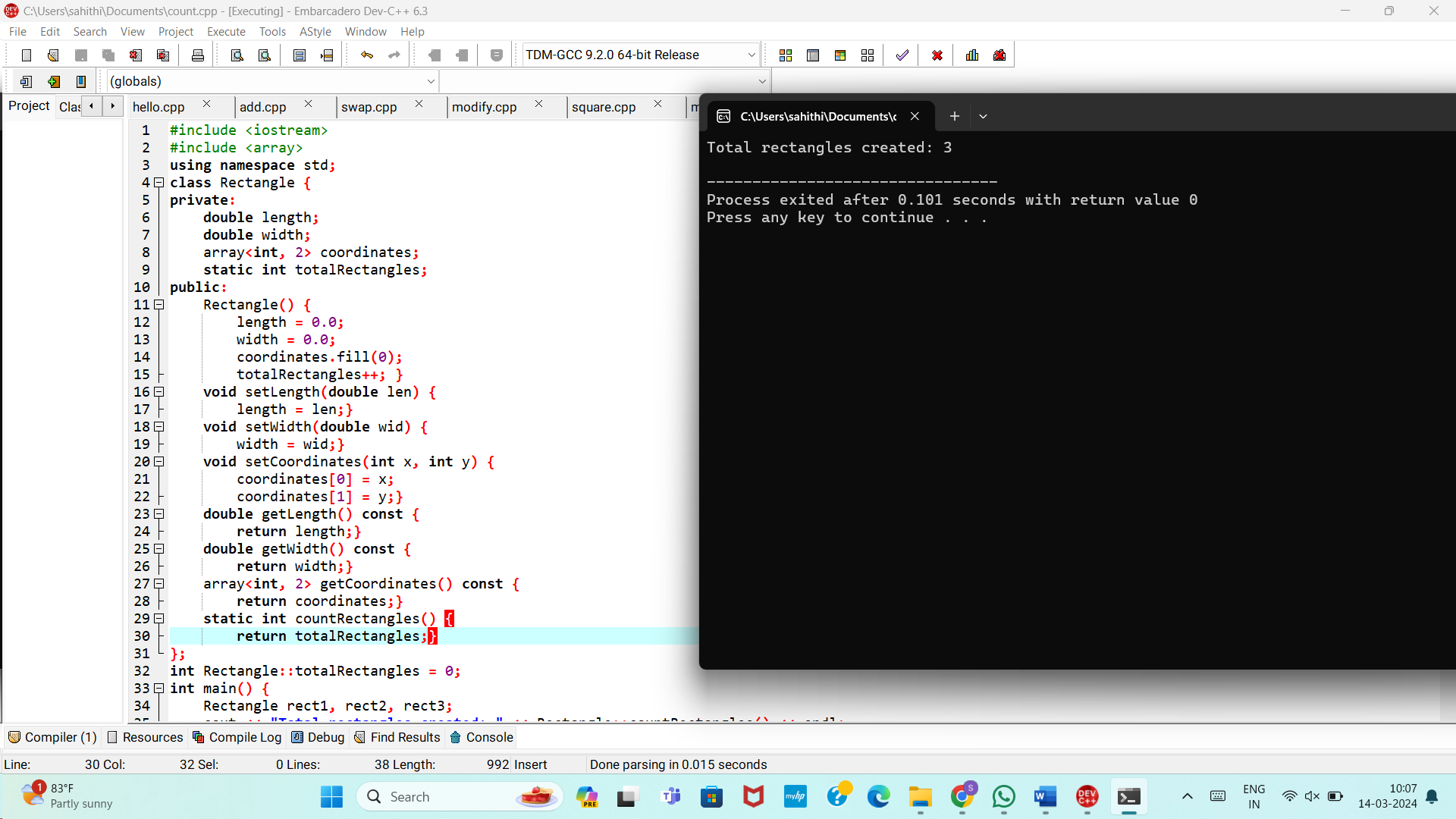
8. Define a C++ class named Rectangle with private member variables length and width. Implement public member functions to set and get the length and width of the rectangle.



9. Extend the Rectangle class to include an array of integers named coordinates to store the (x, y) coordinates of the rectangle. Implement member functions to set and get the coordinates.

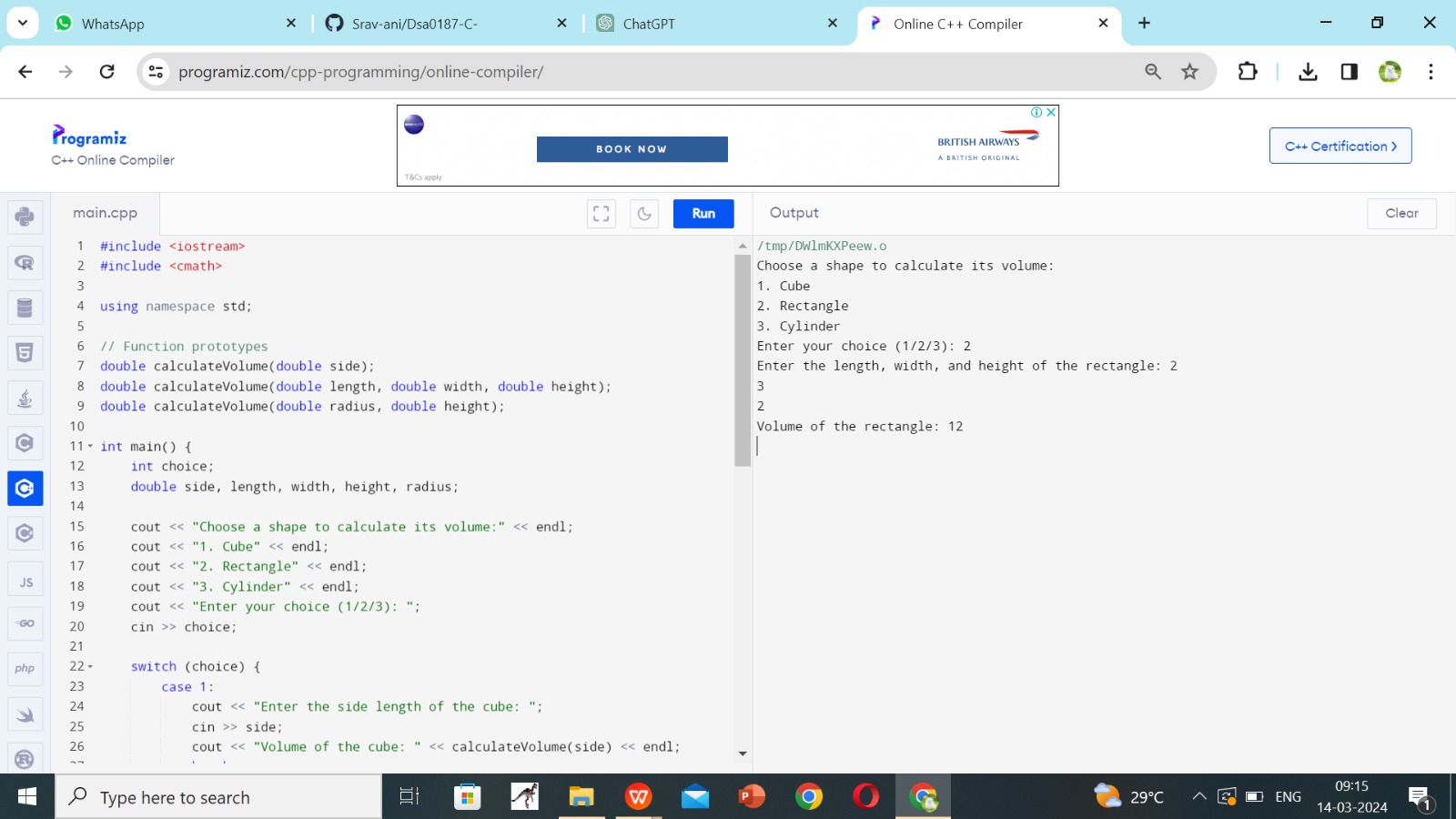


10. Add a static member function named countRectangles to the Rectangle class that keeps track of the total number of rectangle objects created. Display the count in the main function.

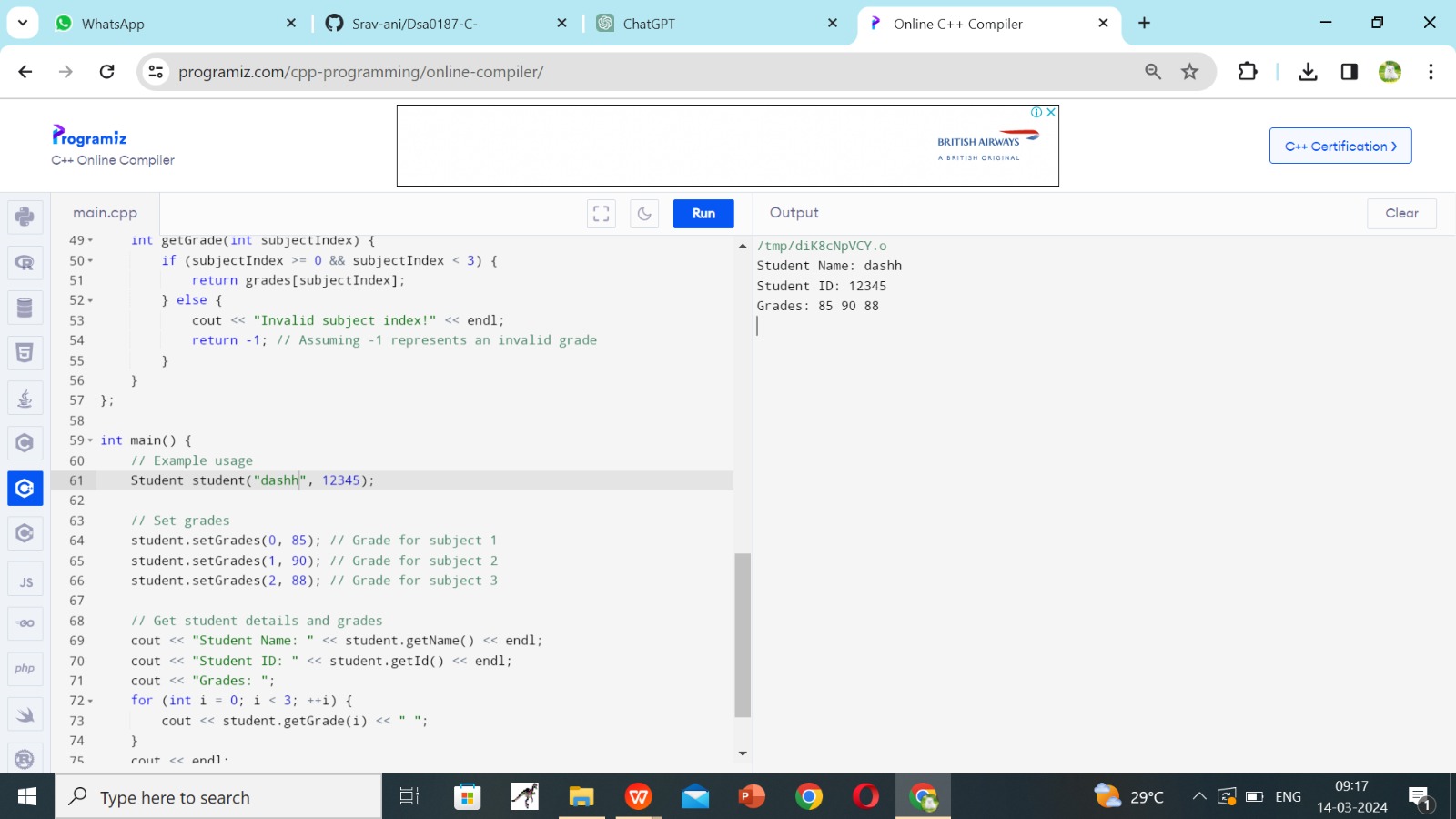


MEDIUM PROGRAMS:

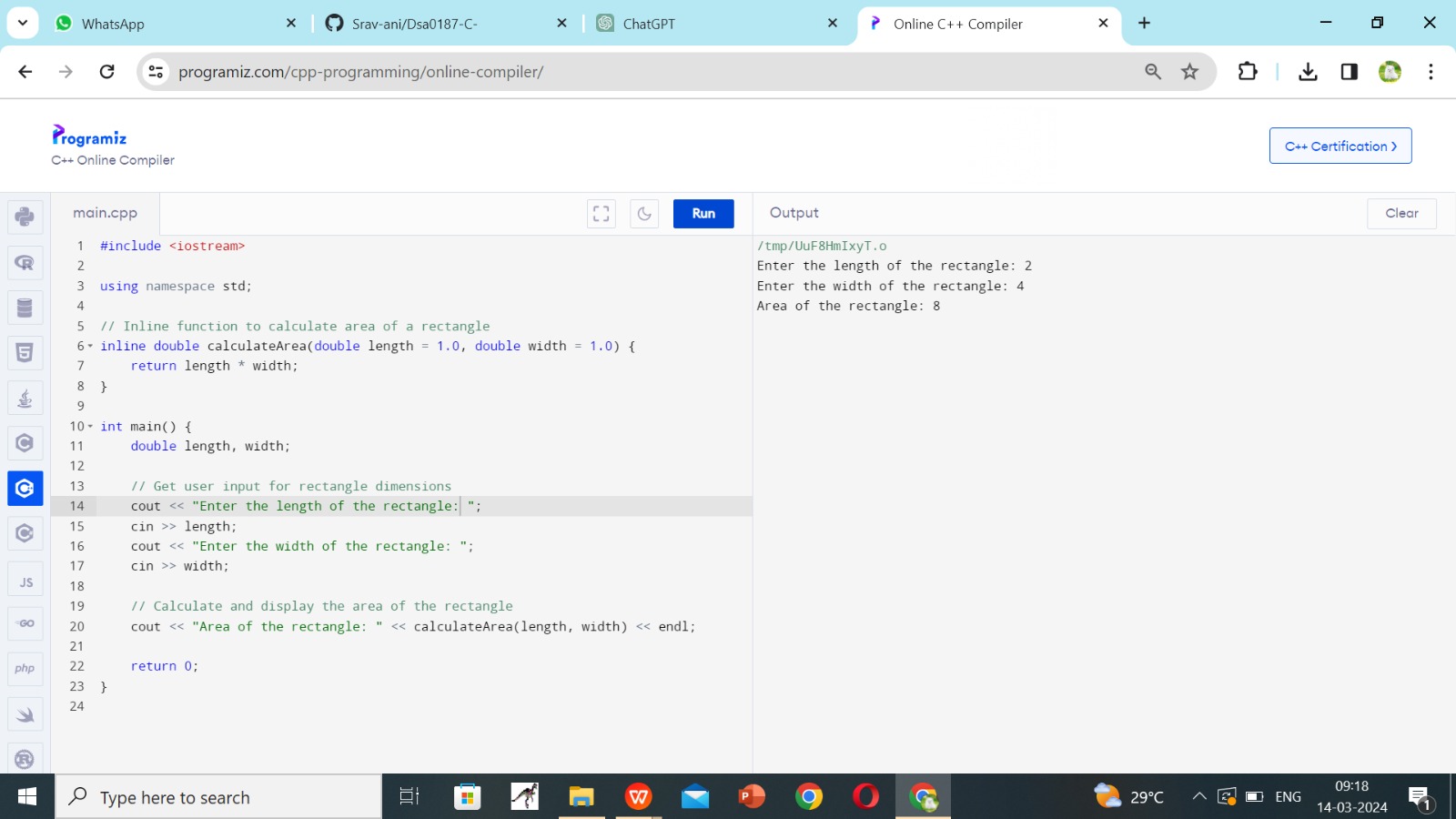
1.Write a C++ program that calculates the volume of a cube, rectangle, or cylinder based on user choice. Use function prototypes and function overloading to define separate functions for each shape's volume calculation.



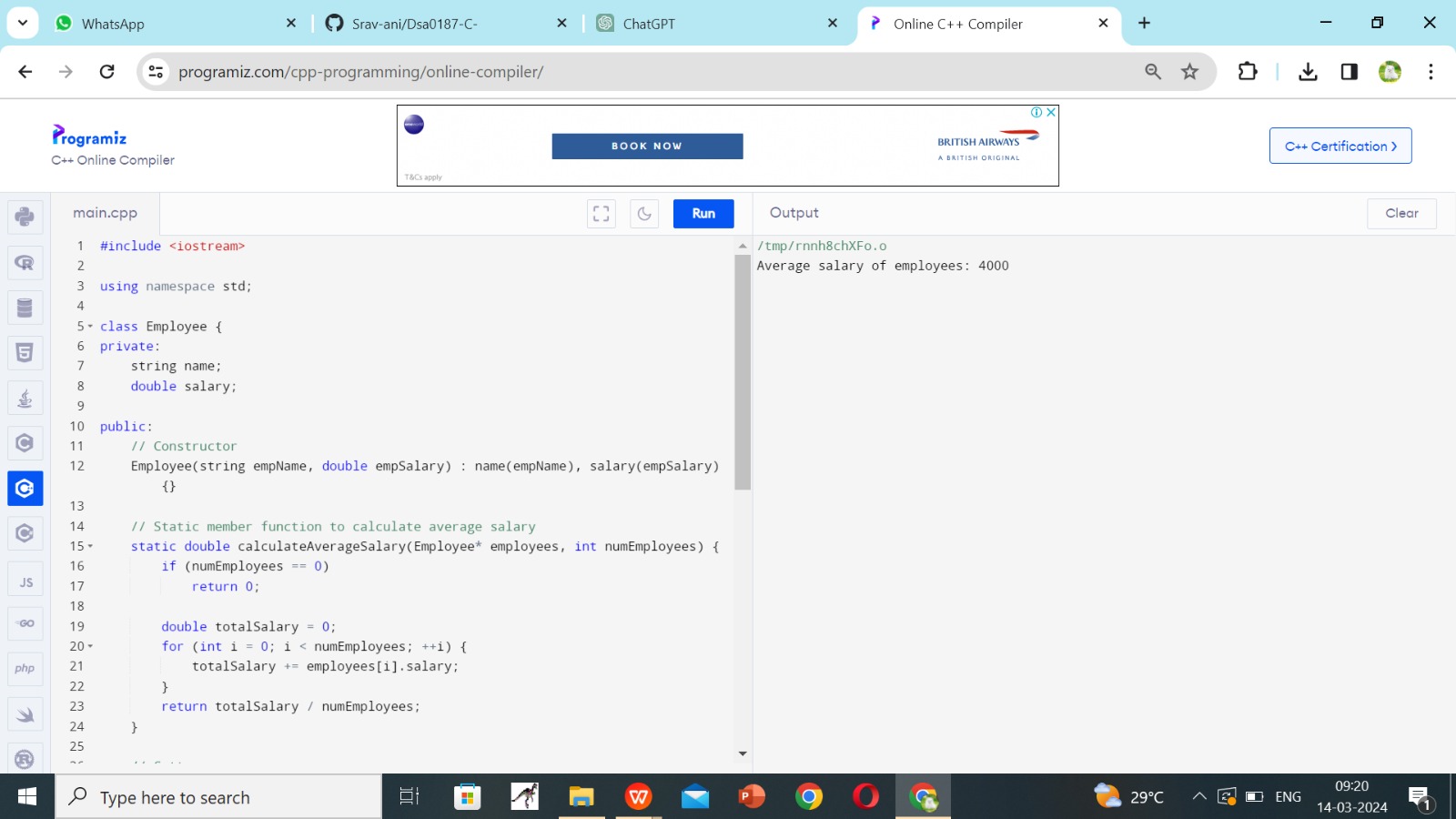
2. Define a class named Student with private member variables name, id, and an array grades to store the student's grades in three subjects. Implement member functions to set and get the student details and grades.



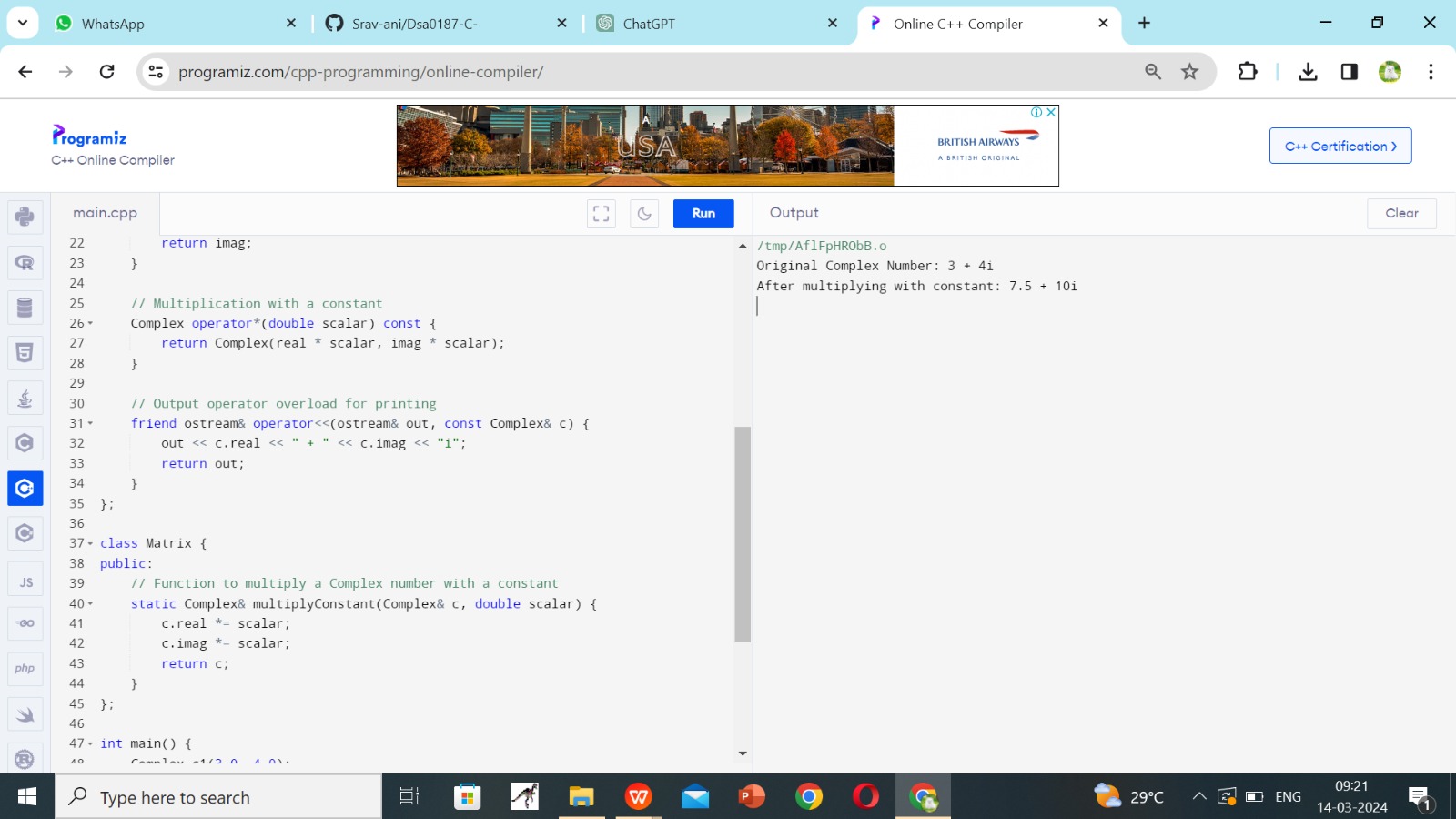
3. Create a C++ program that defines an inline function named calculateArea to calculate the area of a rectangle. Provide default arguments for length and width parameters. Use this function to calculate the area of a rectangle with user-input dimensions.



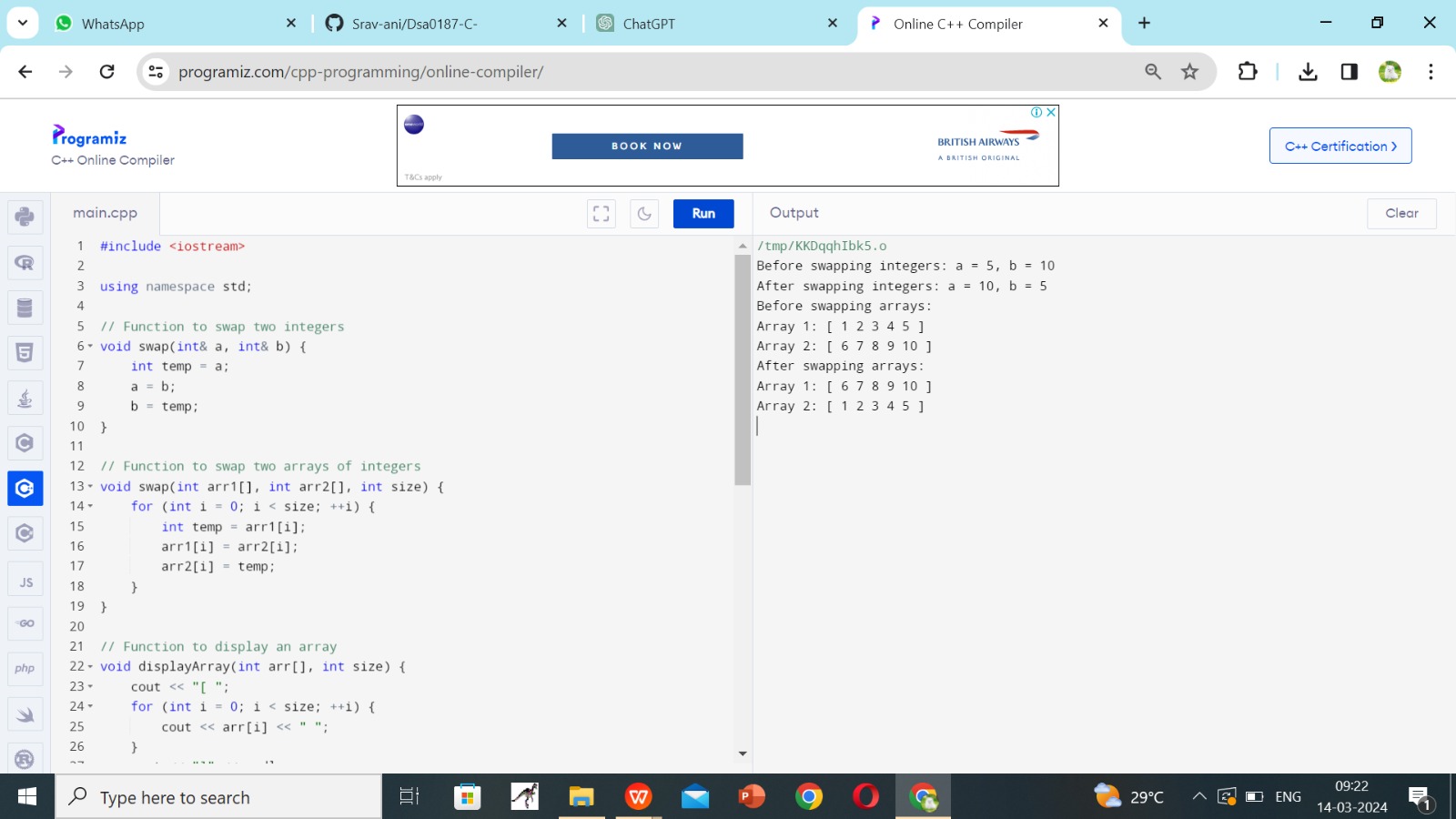
4. Define a class named Employee with private member variables name and salary. Implement a static member function to calculate the average salary of an array of Employee objects.



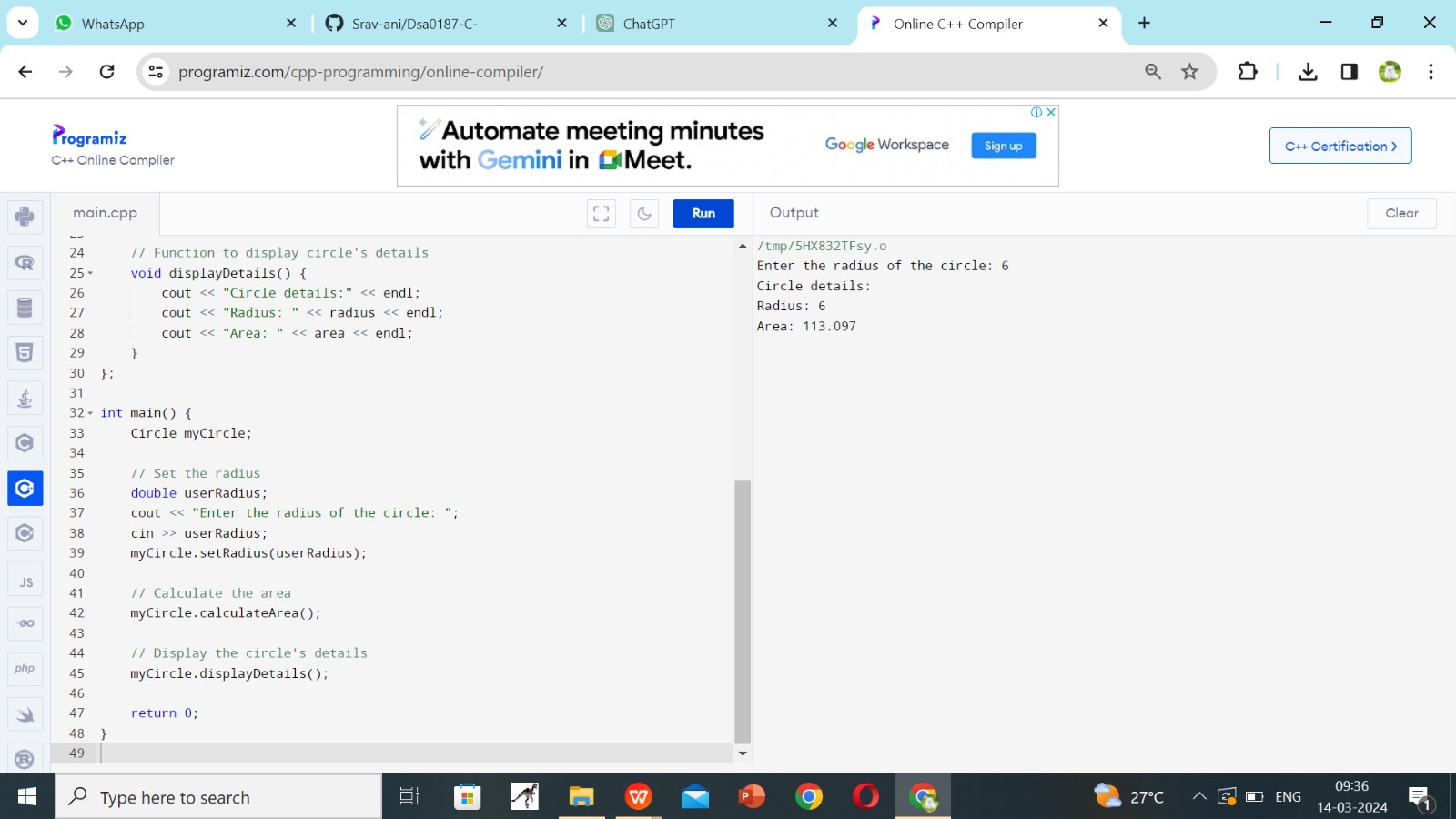
5. Define two classes, Complex and Matrix. Make Matrix a friend of Complex. Implement a function in Matrix class that multiplies a Complex number with a constant and returns the result by reference.



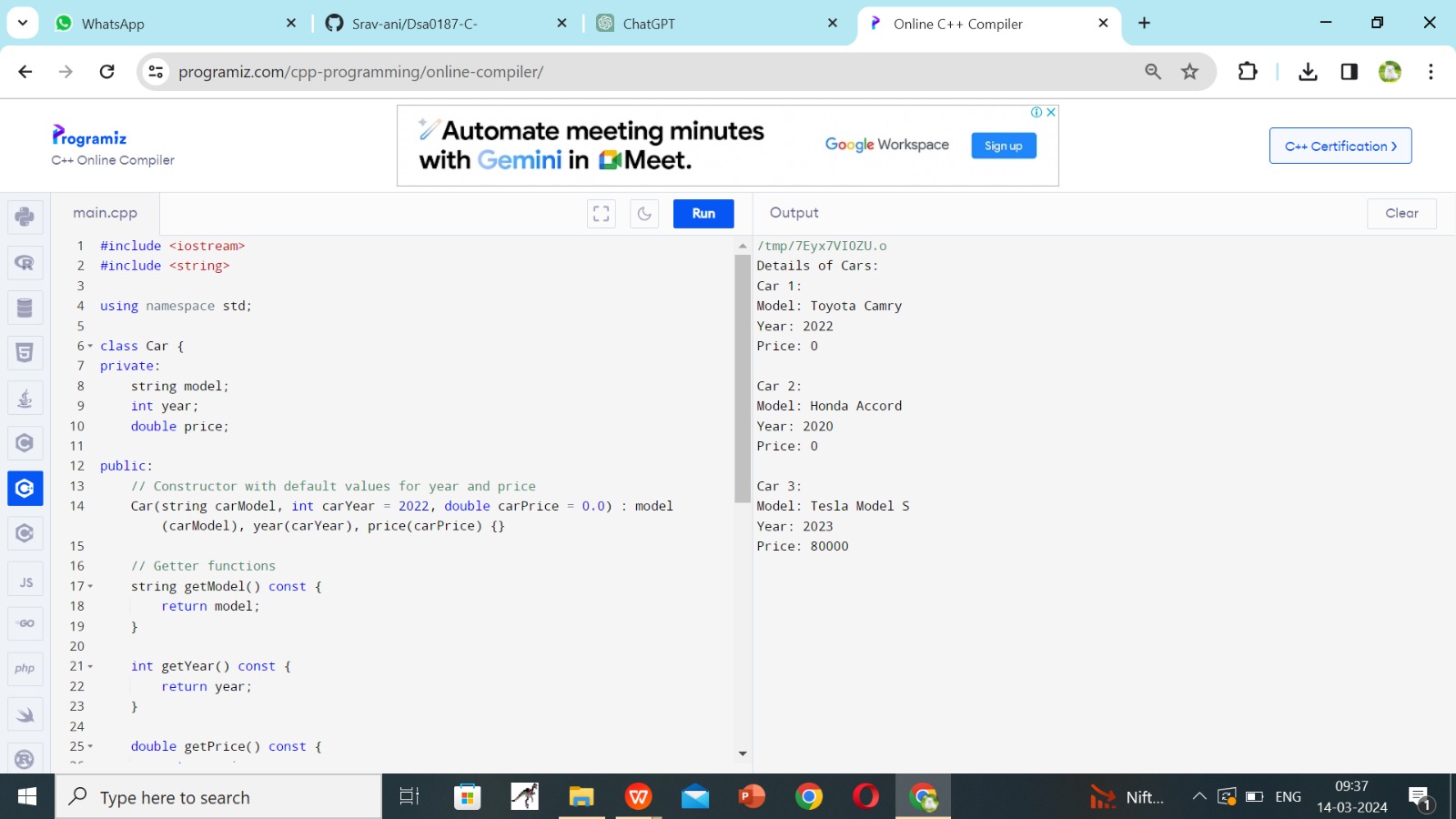
6. Write a C++ program that contains overloaded functions named swap. Define one version that swaps two integers and another version that swaps two arrays of integers. Implement these functions using call by reference.



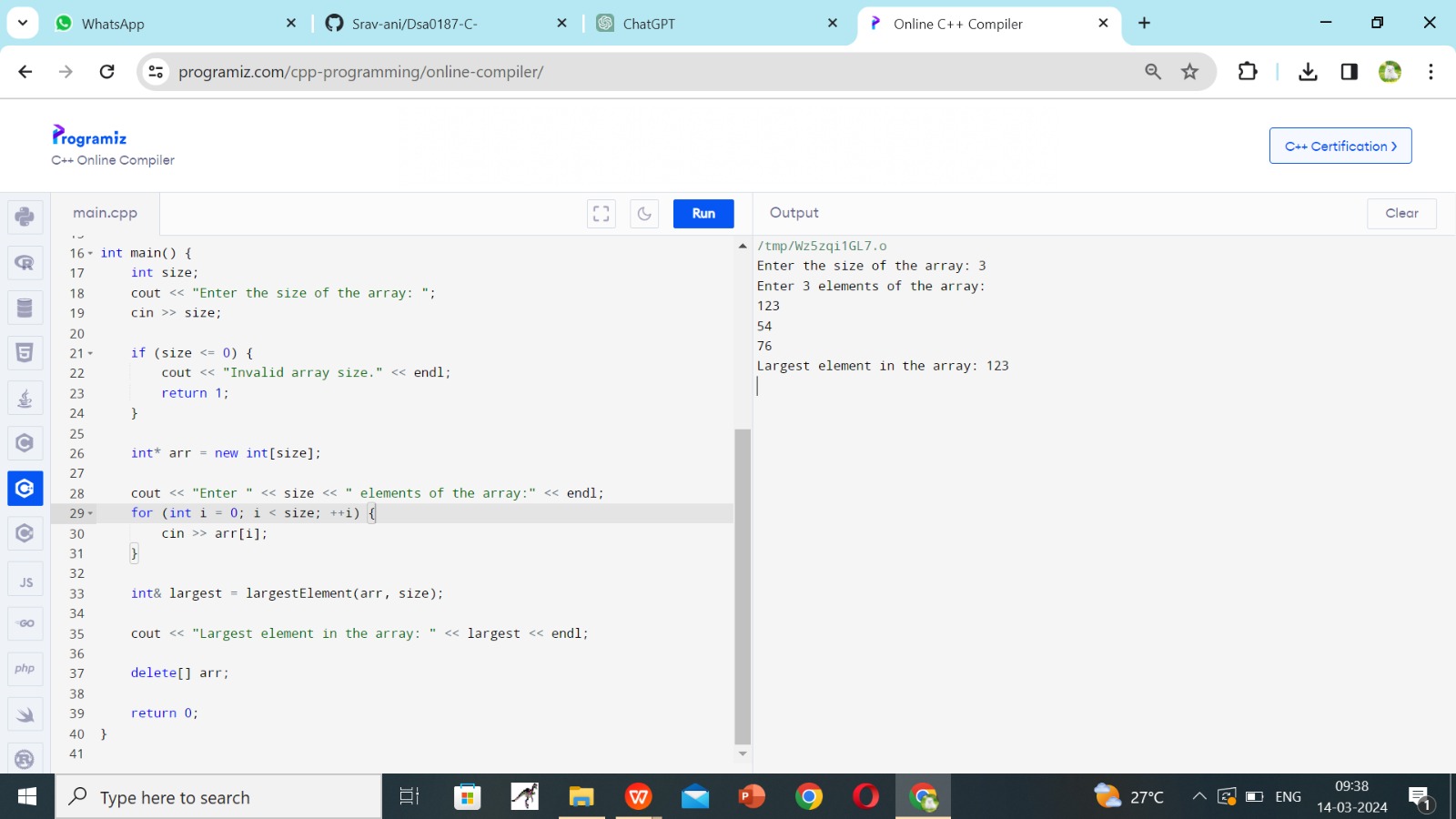
7. Define a class named Circle with private member variables radius and area. Implement member functions to set the radius, calculate the area, and display the circle's details.



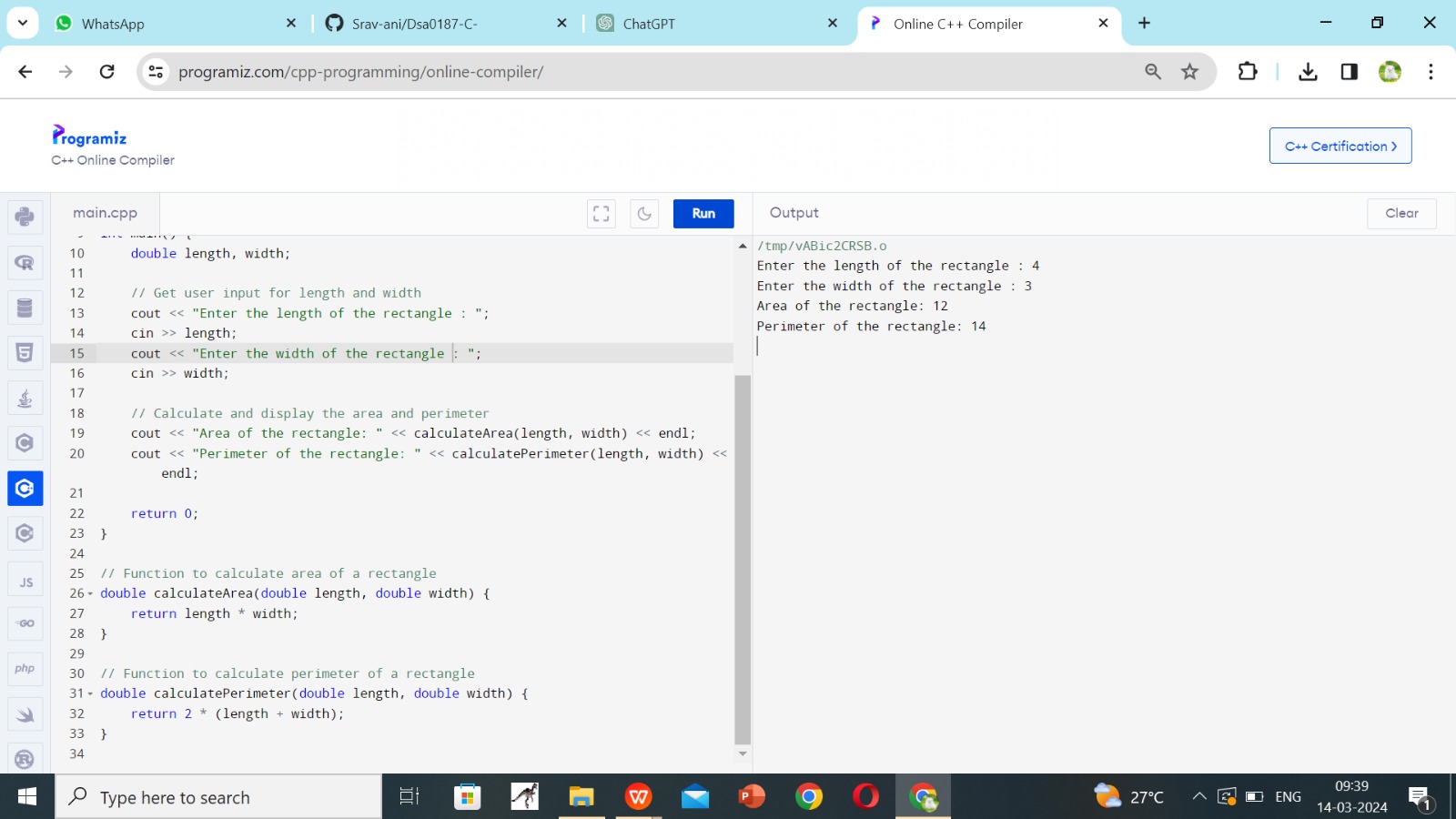
8. Create a C++ program that defines a class named Car with private member variables model, year, and price. Implement an array of Car objects and provide default values for year and price.



9. Define a C++ function named largestElement that returns a reference to the largest element in an array of integers. Use this function to find the largest element in a user-input array.

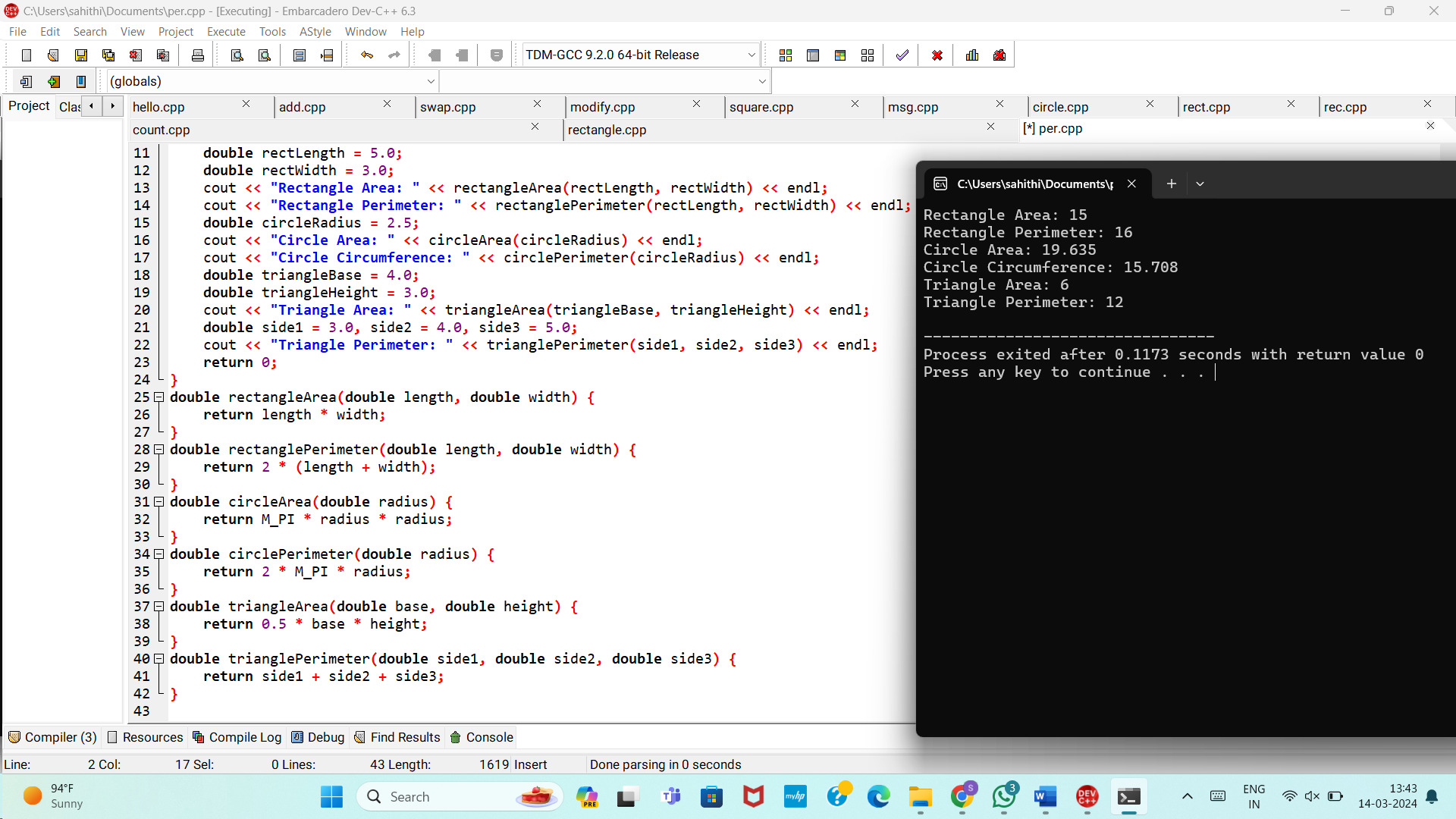


10. Write a C++ program with function prototypes for calculating the area and perimeter r of a rectangle. Implement these functions with default arguments for length and width. Prompt the user to enter the length and width to calculate the area and perimeter.

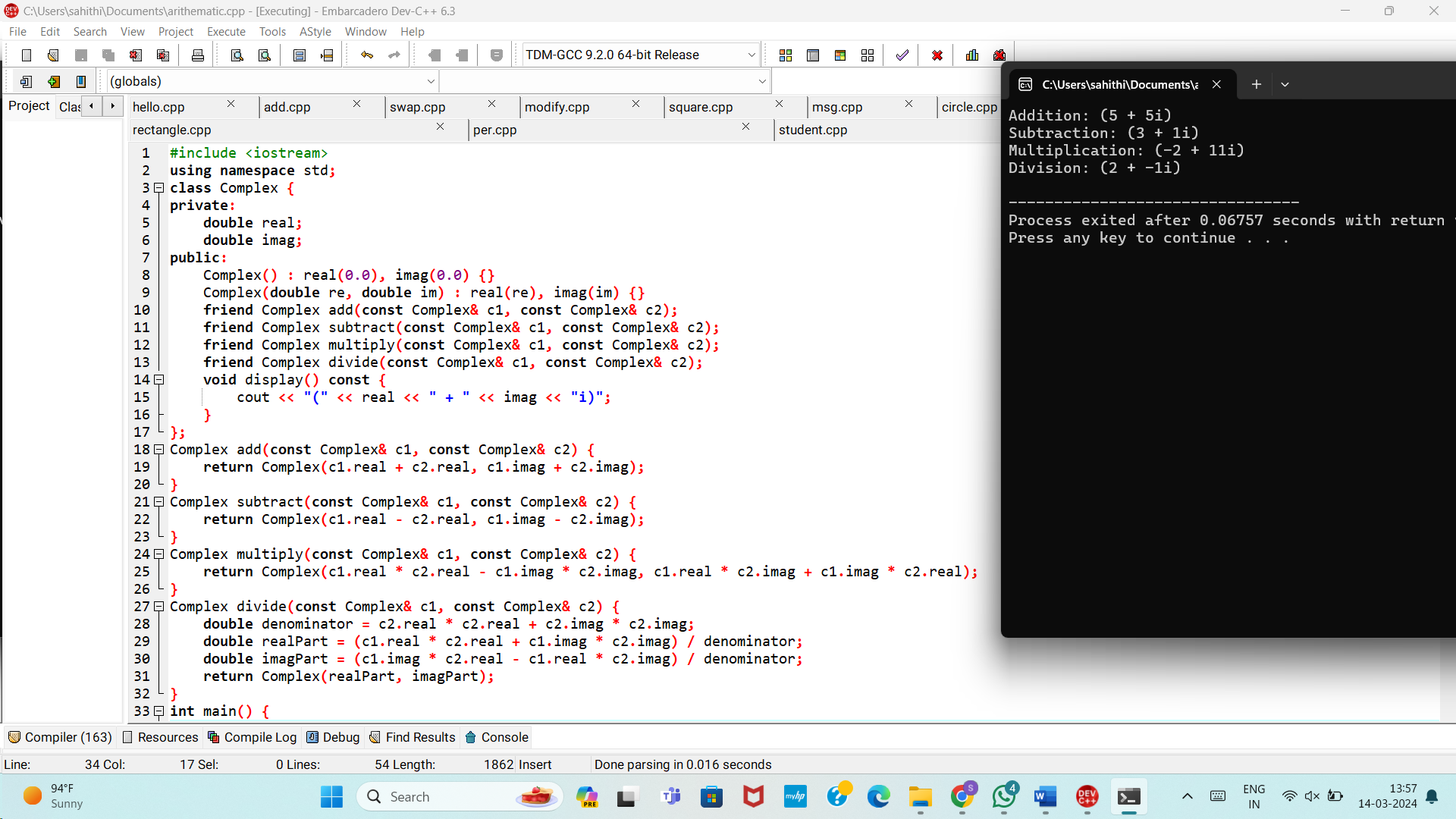


HARD PROGRAMS

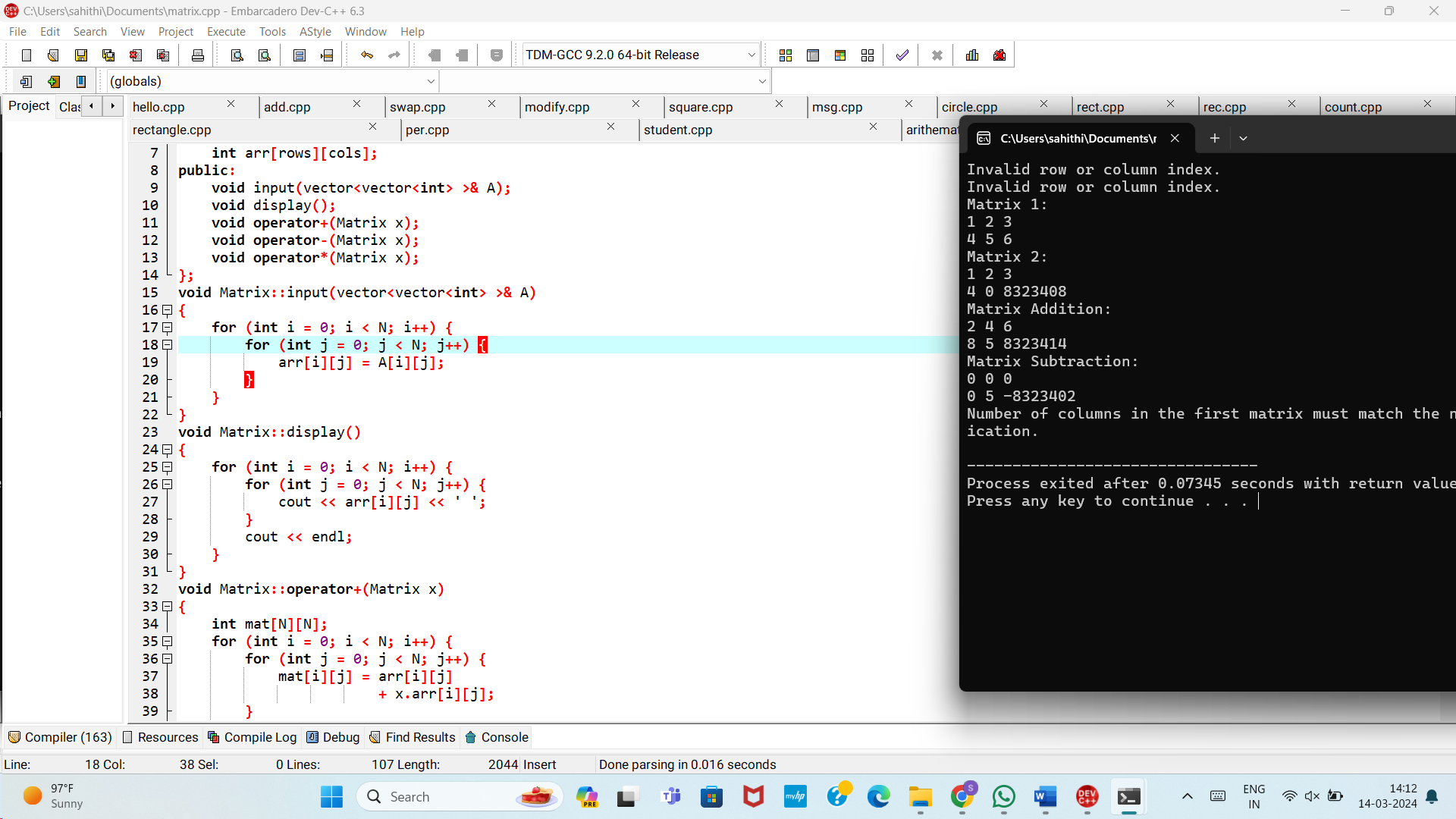
1)Write a C++ program with function prototypes for calculating the area and perimeter of a rectangle, circle, and triangle.



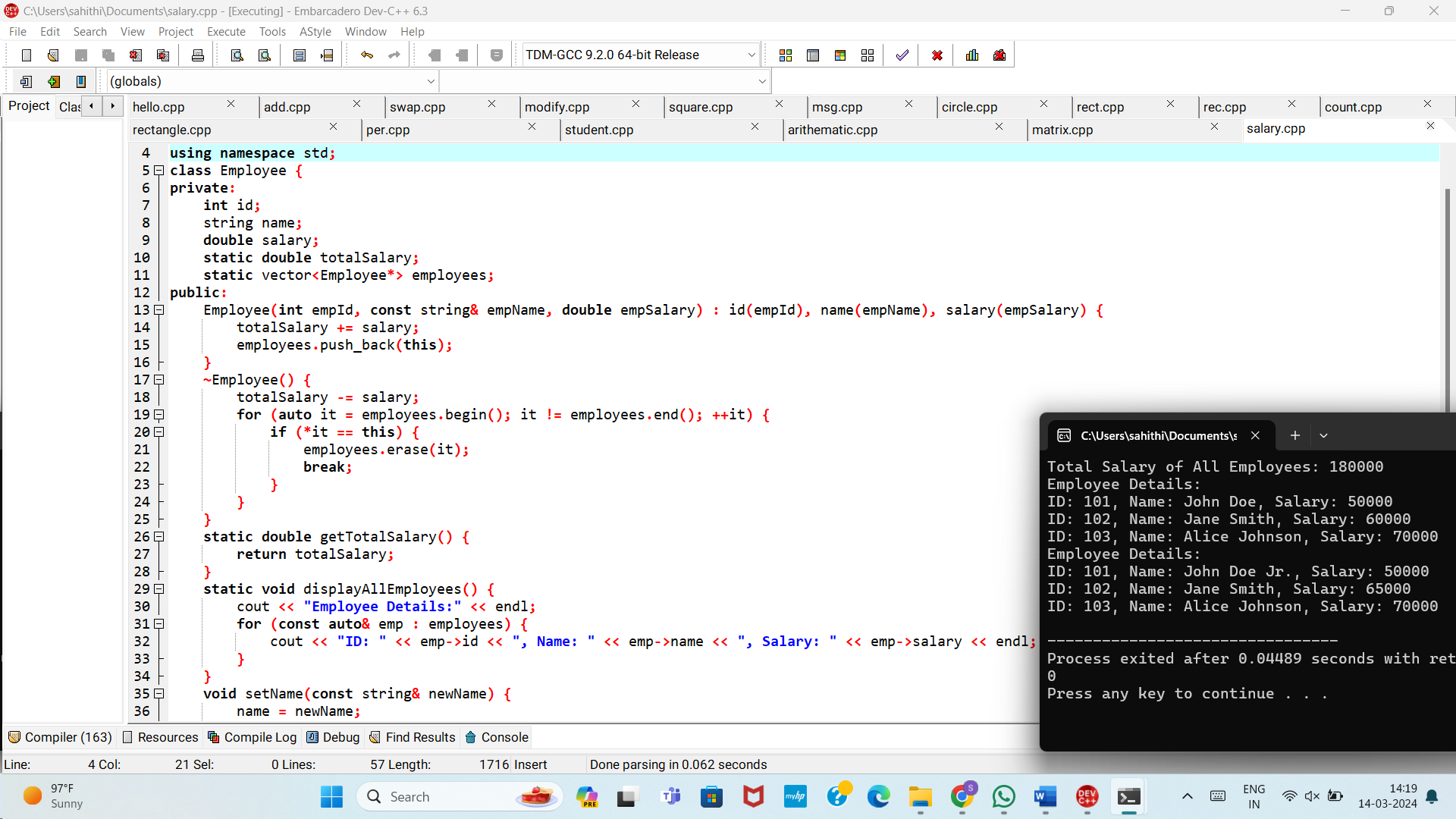
2.Define a class named Complex representing complex numbers with real and imaginary parts. Implement friend functions for addition, subtraction, multiplication, and division of complex numbers.



3.Create a C++ program that defines a class named Matrix to represent a 2D matrix. Implement operator overloading for matrix addition, subtraction, and multiplication. Use dynamic memory allocation for the matrix array within the class.

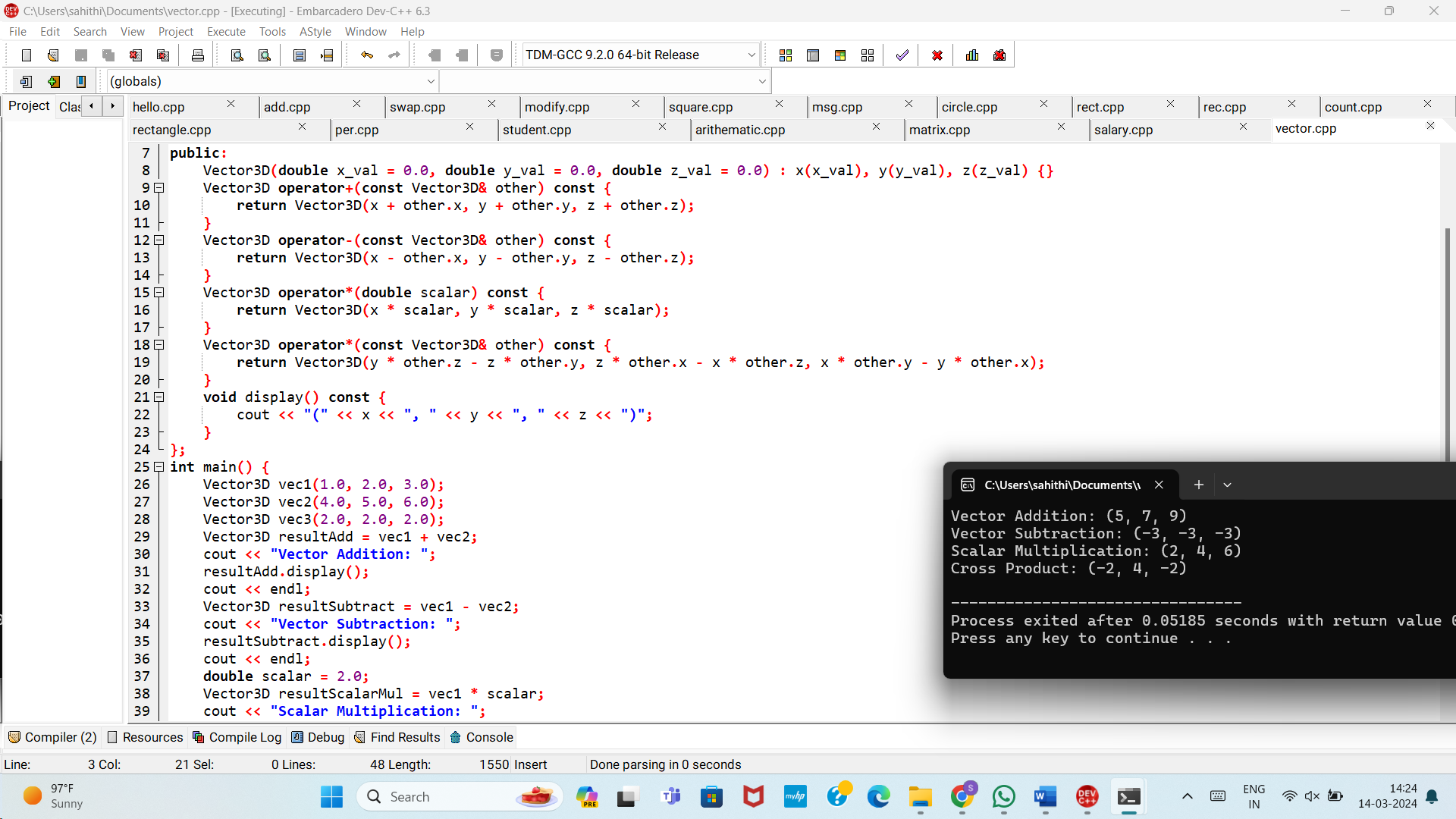


4.Define a class named Employee with private member variables id, name, and salary. Implement a static member function to calculate the total salary of all employees. Include advanced member functions for employee details manipulation.

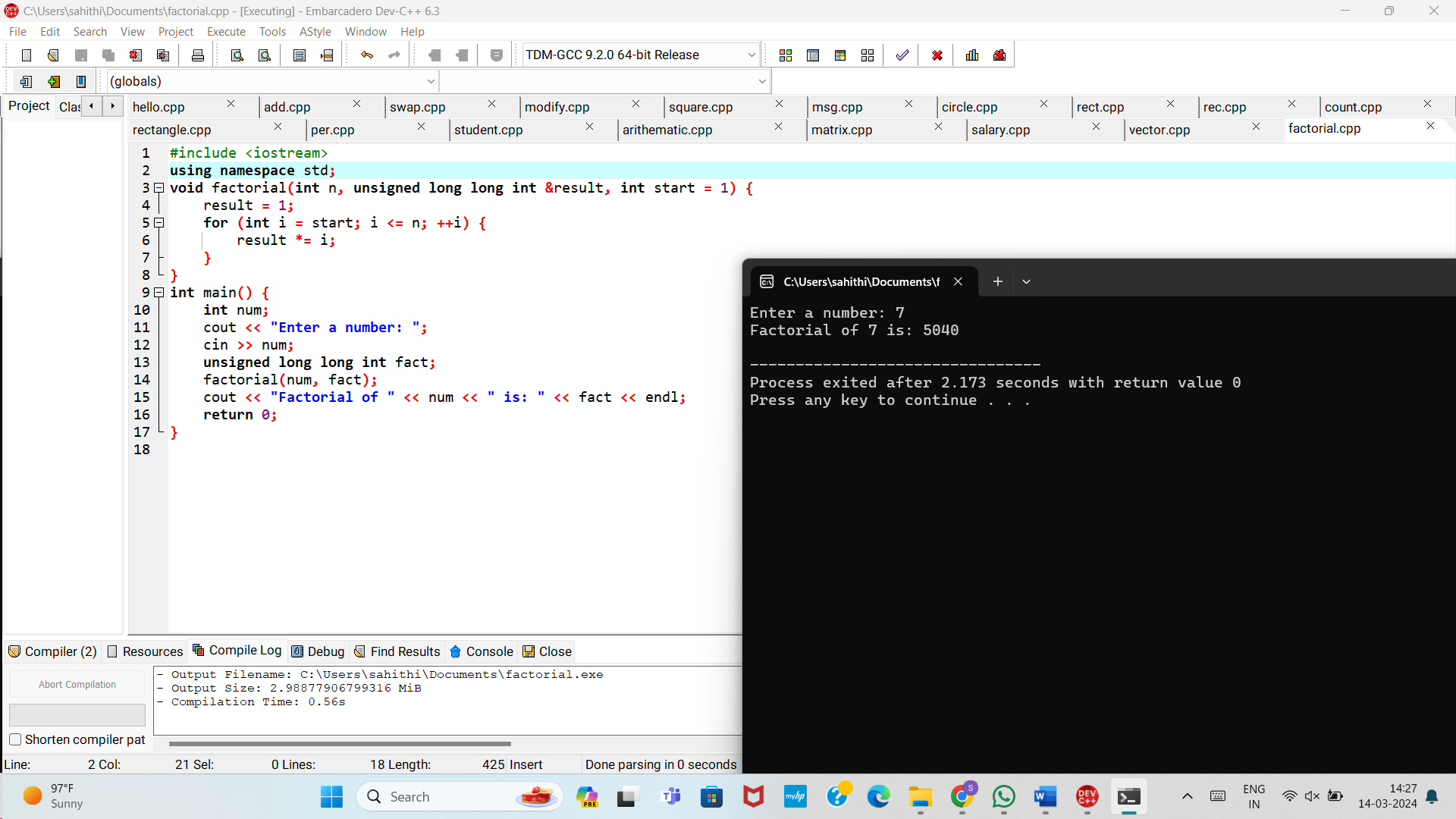


**5.** Develop a C++ program that defines a class named Vector3D representing a 3D vector. Implement operator overloading for addition, subtraction, scalar multiplication, and cross product operations on arrays of Vector3D objects.

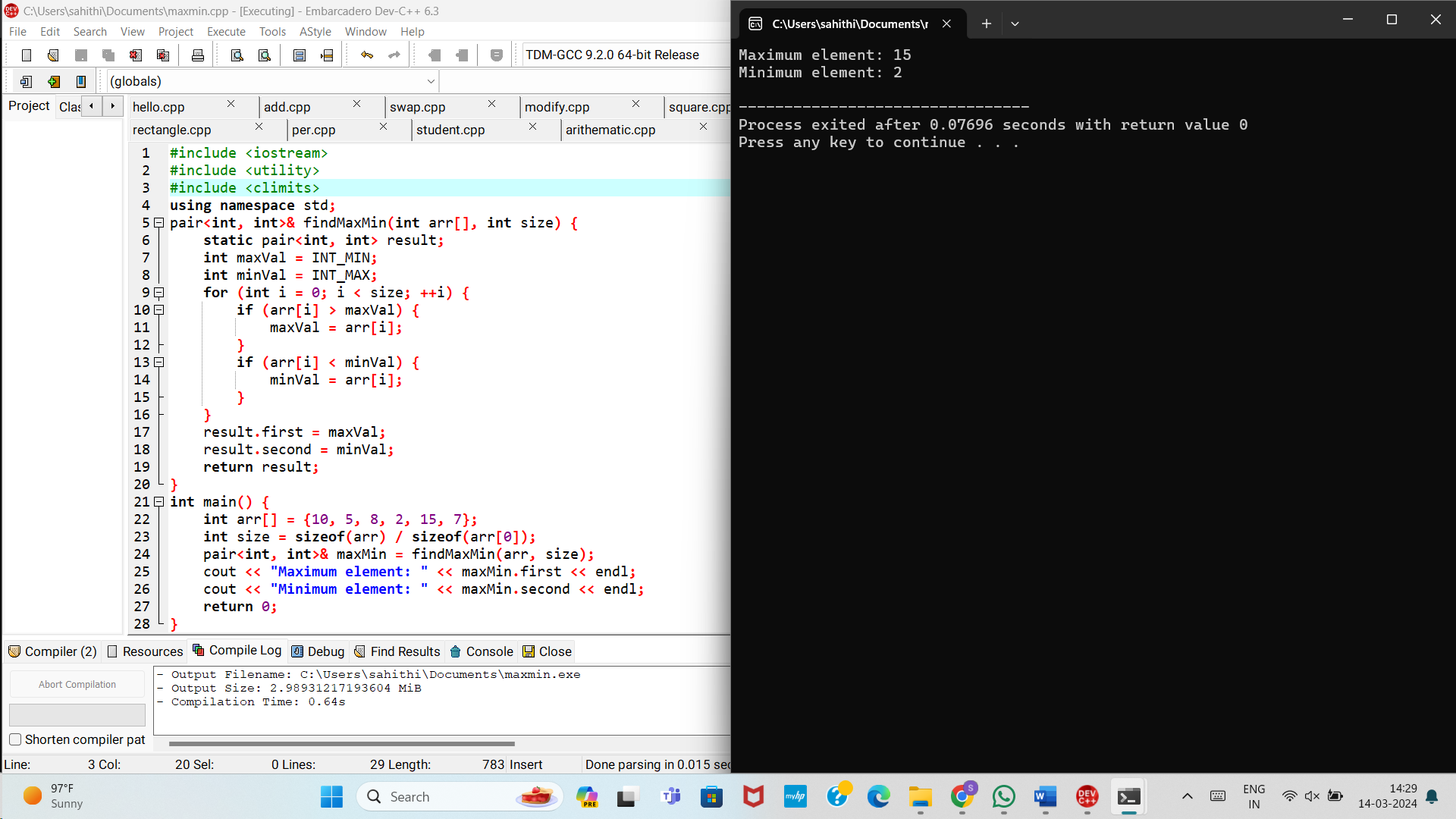
1. 6.Write a C++ program that contains a function to calculate the factorial of a number using call by reference. Use default arguments to provide a starting value for the factorial calculation.



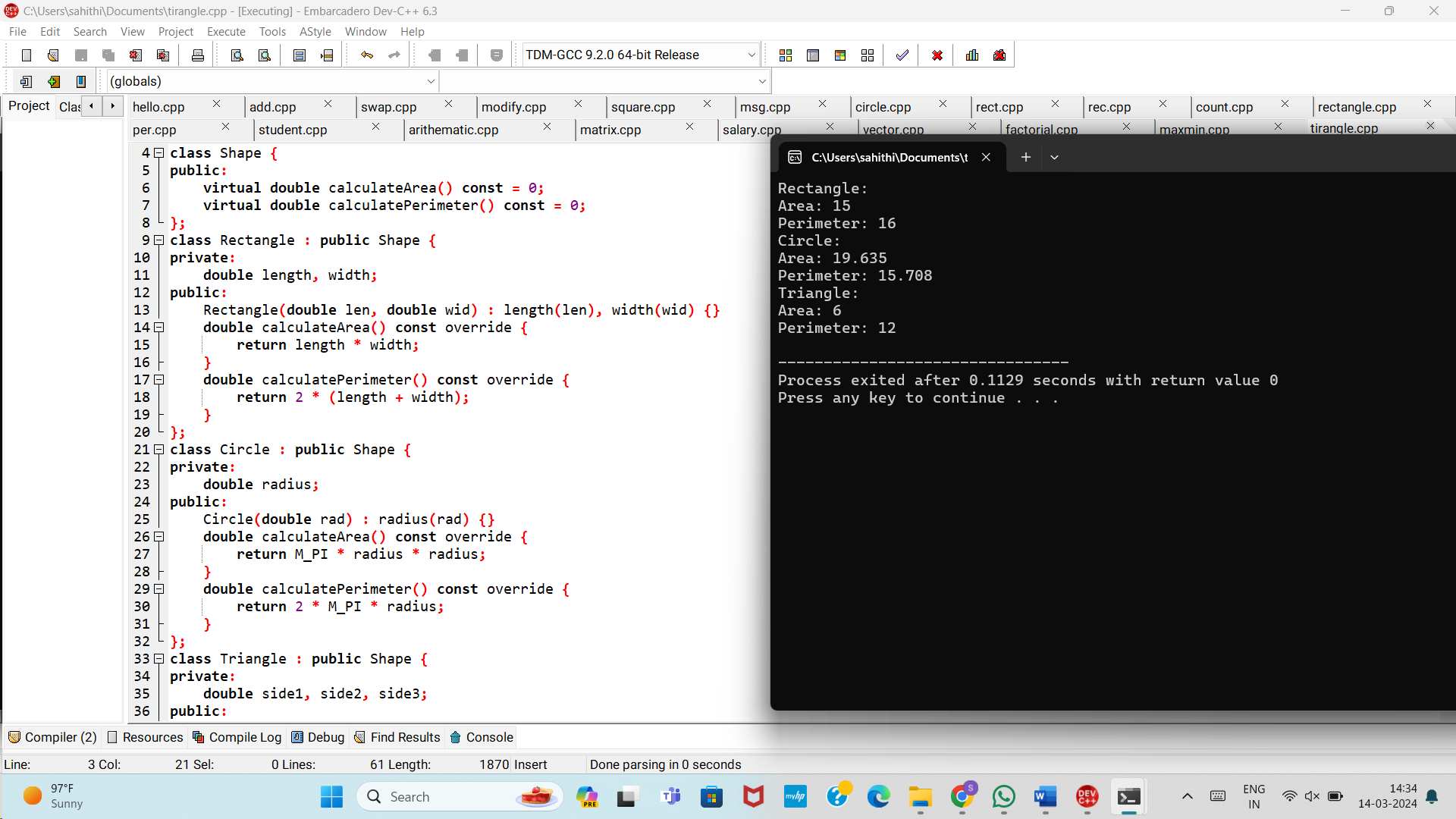
6.Write a C++ program that contains a function to calculate the factorial of a number using call by reference. Use default arguments to provide a starting value for the factorial calculation.



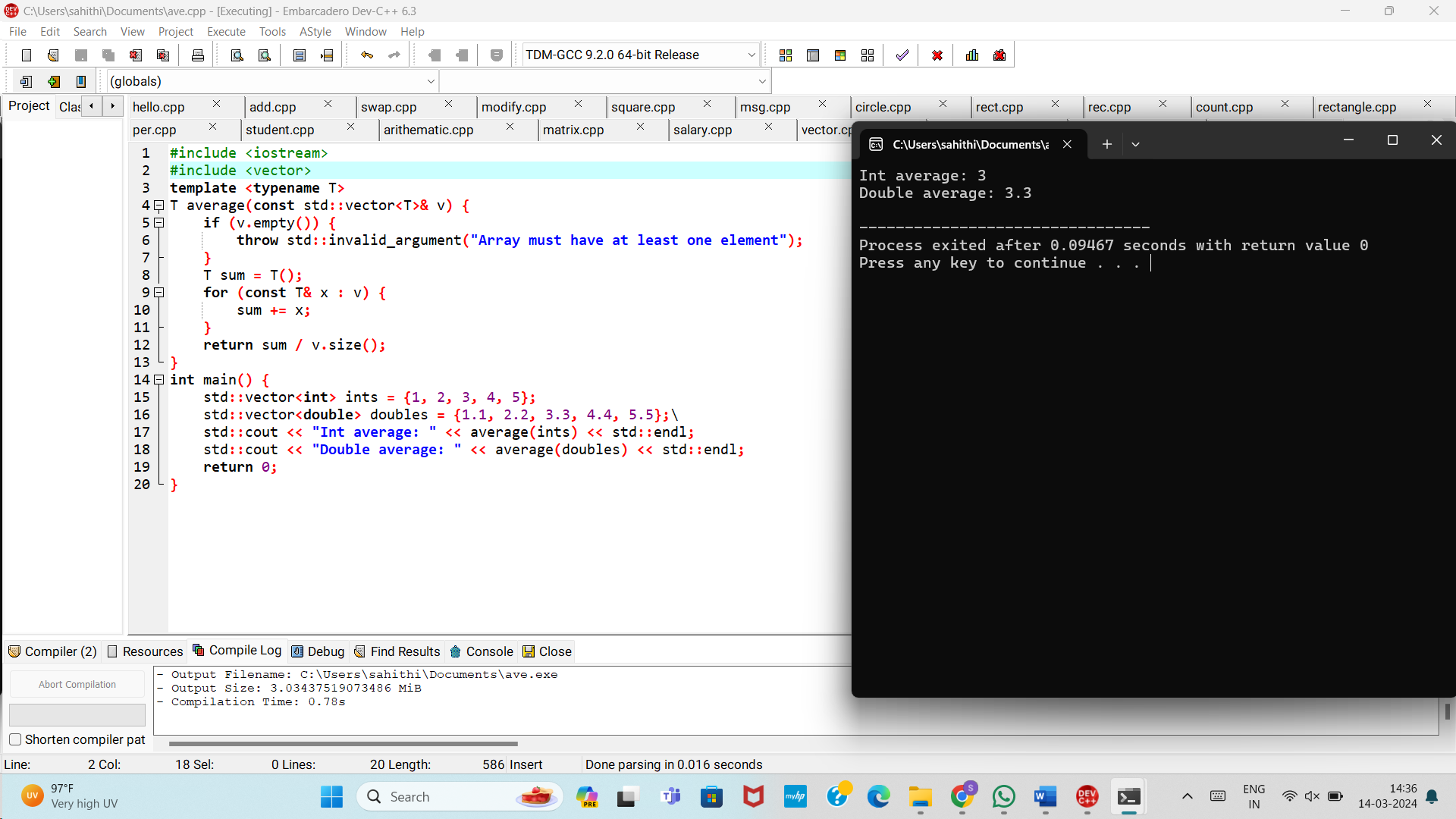
7.Define a C++ function named findMaxMin that finds both the maximum and minimum elements in an array of integers. Return a reference to a pair object containing the maximum and minimum values.



8.Create a C++ program that defines a base class named Shape with virtual member functions for calculating area and perimeter. Implement derived classes for specific shapes like Rectangle, Circle, and Triangle, overriding the virtual functions accordingly.



9.Write a C++ program that defines a template function named average to calculate the average of elements in an array of any data type. Use function prototypes to declare the template function and test it with different data types.



10.Define a class named Pair representing a pair of values of any data type. Implement a friend function to swap the values of two Pair objects. Use template specialization to handle swapping for specific data types.

