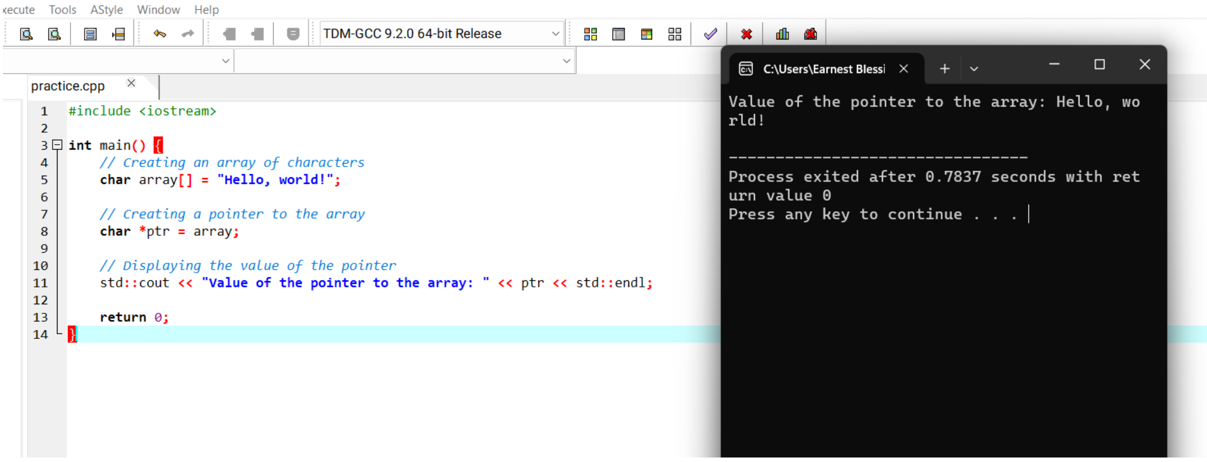
MODEL PRACTICAL EXAM

DSA-0179 OBJECT ORIENTED PROGRAMMING FOR C++

B.Earnest Blessing

192110302

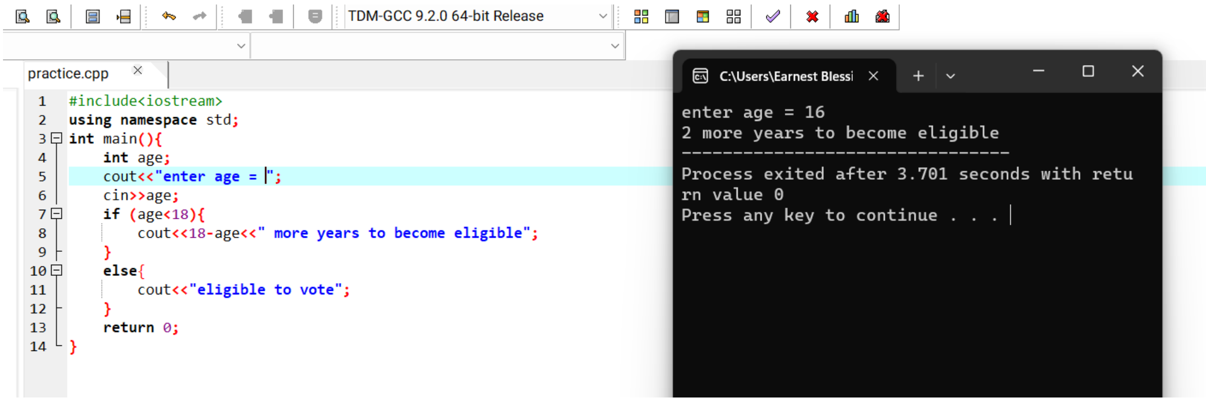
1. C++ program to create pointer to an array of characters and display it's value



```
#include <iostream>

int main() {
    // Creating an array of characters
    char array[] = "Hello, world!";

    // Creating a pointer to the array
    char *ptr = array;

    // Displaying the value of the pointer
    std::cout << "Value of the pointer to the array: " << ptr << std::endl;

    return 0;
}
```

```
Value of the pointer to the array: Hello, wo
rld!

--------------------------------
Process exited after 0.7837 seconds with ret
urn value 0
Press any key to continue . . .
```

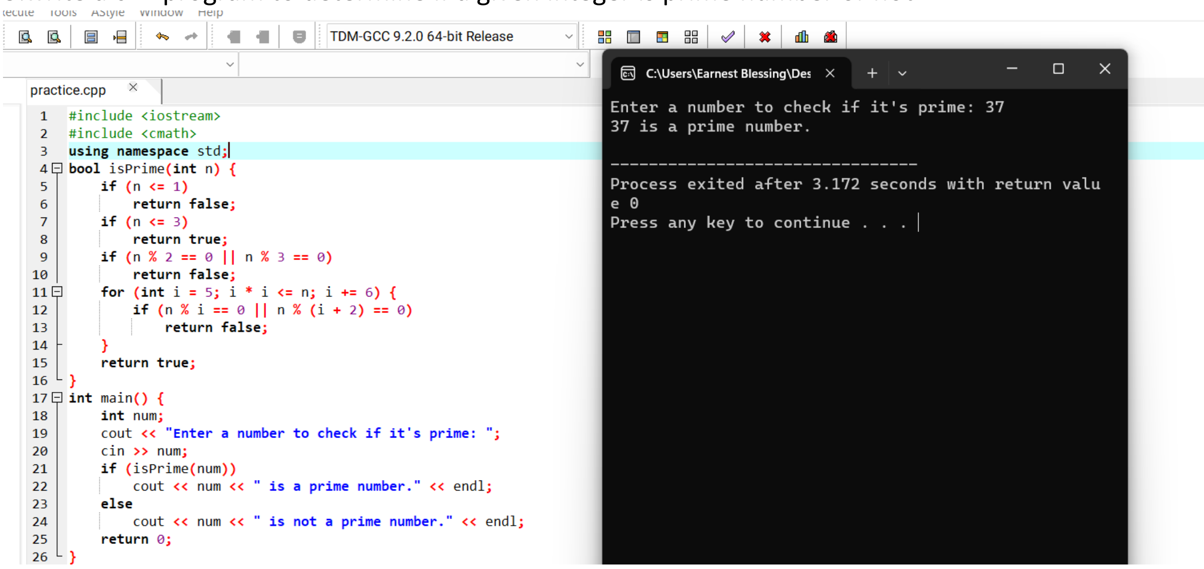
2.Develop a program to find whether the to vote or not and if not then print how many years are left be eligible



```
#include<iostream>
using namespace std;
int main(){
    int age;
    cout<<"enter age = ";
    cin>>age;
    if (age<18){
        cout<<18-age<<" more years to become eligible";
    }
    else{
        cout<<"eligible to vote";
    }
    return 0;
}
```

```
enter age = 16
2 more years to become eligible
--------------------------------
Process exited after 3.701 seconds with retu
rn value 0
Press any key to continue . . .
```

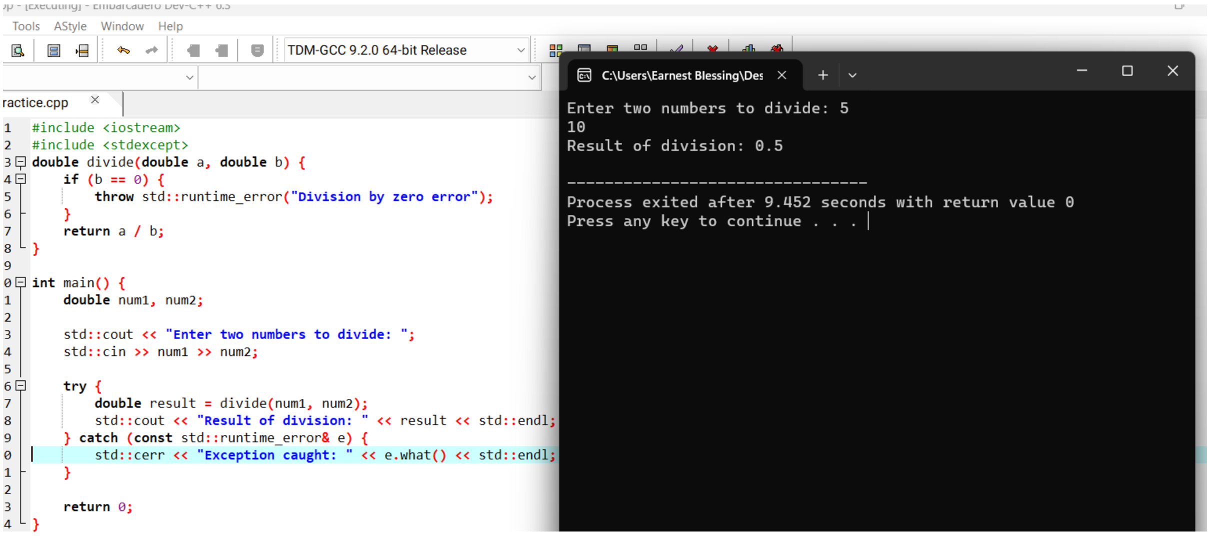
3.write a c++ program to determine if a given integer is prime number or not


xecute Tools AStyle Window Help
TDM-GCC 9.2.0 64-bit Release
practice.cpp
C:\Users\Earnest Blessing\Des

```
#include <iostream>
#include <cmath>
using namespace std;
bool isPrime(int n) {
    if (n <= 1)
        return false;
    if (n <= 3)
        return true;
    if (n % 2 == 0 || n % 3 == 0)
        return false;
    for (int i = 5; i * i <= n; i += 6) {
        if (n % i == 0 || n % (i + 2) == 0)
            return false;
    }
    return true;
}
int main() {
    int num;
    cout << "Enter a number to check if it's prime: ";
    cin >> num;
    if (isPrime(num))
        cout << num << " is a prime number." << endl;
    else
        cout << num << " is not a prime number." << endl;
    return 0;
}
```

```
Enter a number to check if it's prime: 37
37 is a prime number.

--------------------------------
Process exited after 3.172 seconds with return valu
e 0
Press any key to continue . . .
```

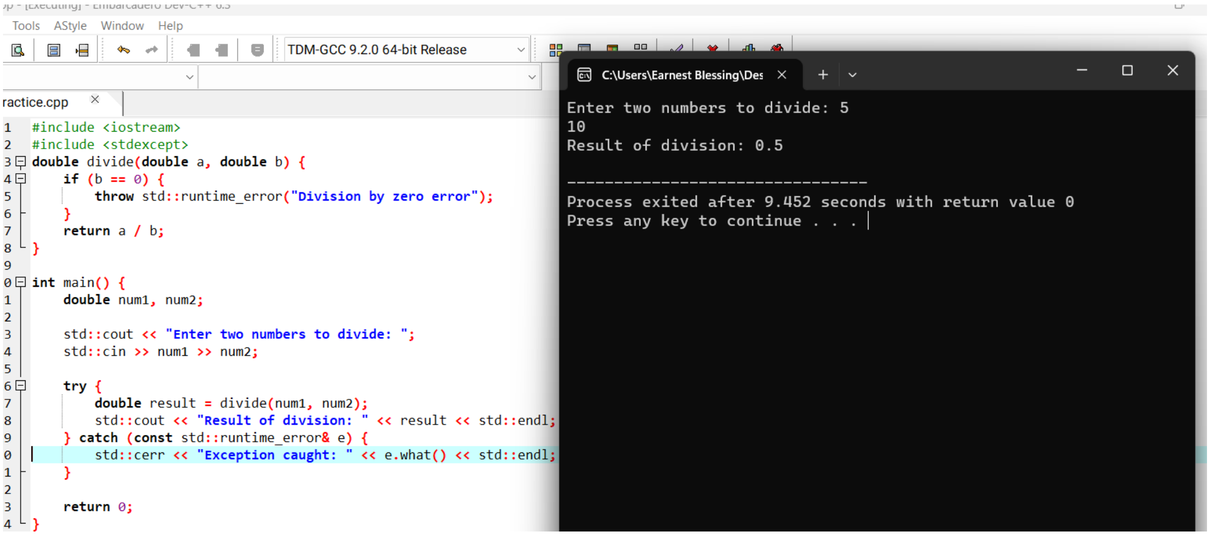
## 4. Program to demonstrate the use of standard exception class for handling exception



```
#include <iostream>
#include <stdexcept>
double divide(double a, double b) {
    if (b == 0) {
        throw std::runtime_error("Division by zero error");
    }
    return a / b;
}

int main() {
    double num1, num2;

    std::cout << "Enter two numbers to divide: ";
    std::cin >> num1 >> num2;

    try {
        double result = divide(num1, num2);
        std::cout << "Result of division: " << result << std::endl;
    } catch (const std::runtime_error& e) {
        std::cerr << "Exception caught: " << e.what() << std::endl;
    }

    return 0;
}
```

```
Enter two numbers to divide: 5
10
Result of division: 0.5

--------------------------------
Process exited after 9.452 seconds with return value 0
Press any key to continue . . .
```

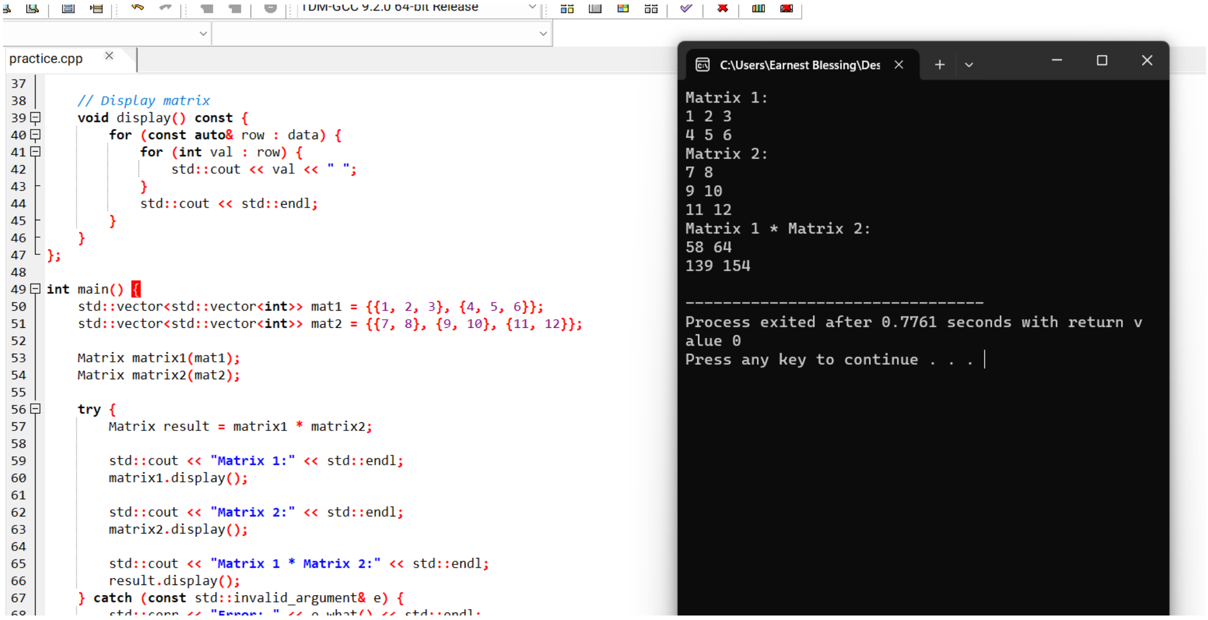
## 5.write a c++ program to overload the -operator to subract two complex numbers



```
#include <iostream>
#include <stdexcept>
double divide(double a, double b) {
    if (b == 0) {
        throw std::runtime_error("Division by zero error");
    }
    return a / b;
}

int main() {
    double num1, num2;

    std::cout << "Enter two numbers to divide: ";
    std::cin >> num1 >> num2;

    try {
        double result = divide(num1, num2);
        std::cout << "Result of division: " << result << std::endl;
    } catch (const std::runtime_error& e) {
        std::cerr << "Exception caught: " << e.what() << std::endl;
    }

    return 0;
}
```

```
Enter two numbers to divide: 5
10
Result of division: 0.5

--------------------------------
Process exited after 9.452 seconds with return value 0
Press any key to continue . . .
```

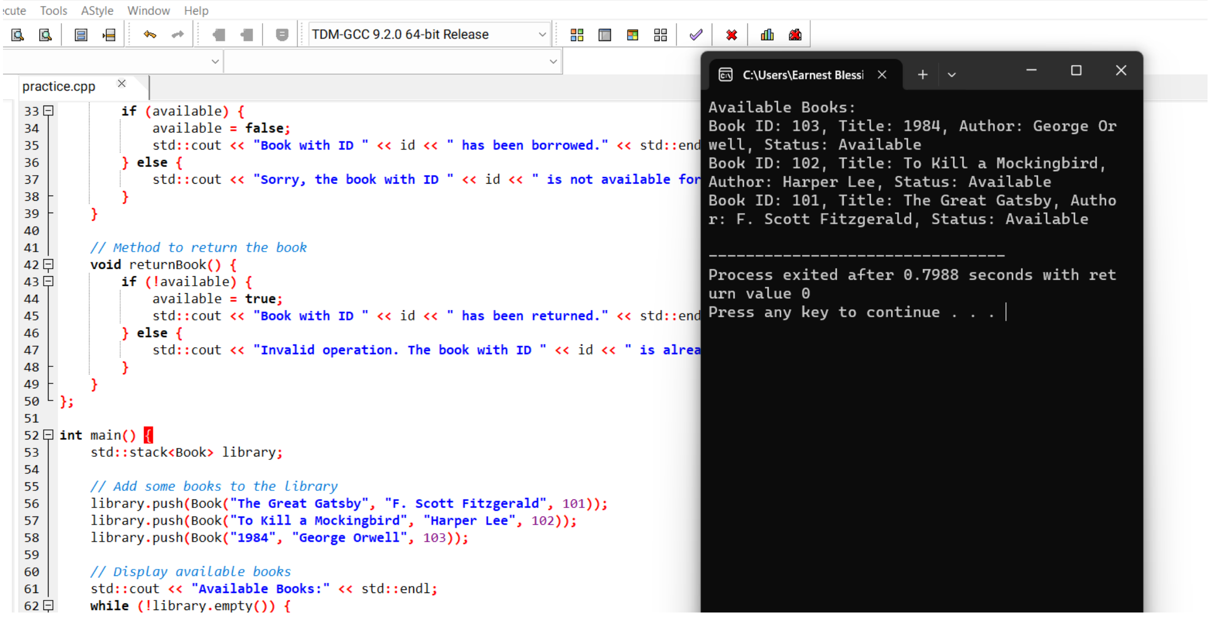
## 6. C++ program to overload the *operator to multiply two matrices



```

    // Display matrix
    void display() const {
        for (const auto& row : data) {
            for (int val : row) {
                std::cout << val << " ";
            }
            std::cout << std::endl;
        }
    }
};

int main() {
    std::vector<std::vector<int>> mat1 = {{1, 2, 3}, {4, 5, 6}};
    std::vector<std::vector<int>> mat2 = {{7, 8}, {9, 10}, {11, 12}};

    Matrix matrix1(mat1);
    Matrix matrix2(mat2);

    try {
        Matrix result = matrix1 * matrix2;

        std::cout << "Matrix 1:" << std::endl;
        matrix1.display();

        std::cout << "Matrix 2:" << std::endl;
        matrix2.display();

        std::cout << "Matrix 1 * Matrix 2:" << std::endl;
        result.display();
    } catch (const std::invalid_argument& e) {
        std::cerr << "Error: " << e.what() << std::endl;
```

```
Matrix 1:
1 2 3
4 5 6
Matrix 2:
7 8
9 10
11 12
Matrix 1 * Matrix 2:
58 64
139 154

--------------------------------
Process exited after 0.7761 seconds with return v
alue 0
Press any key to continue . . .
```

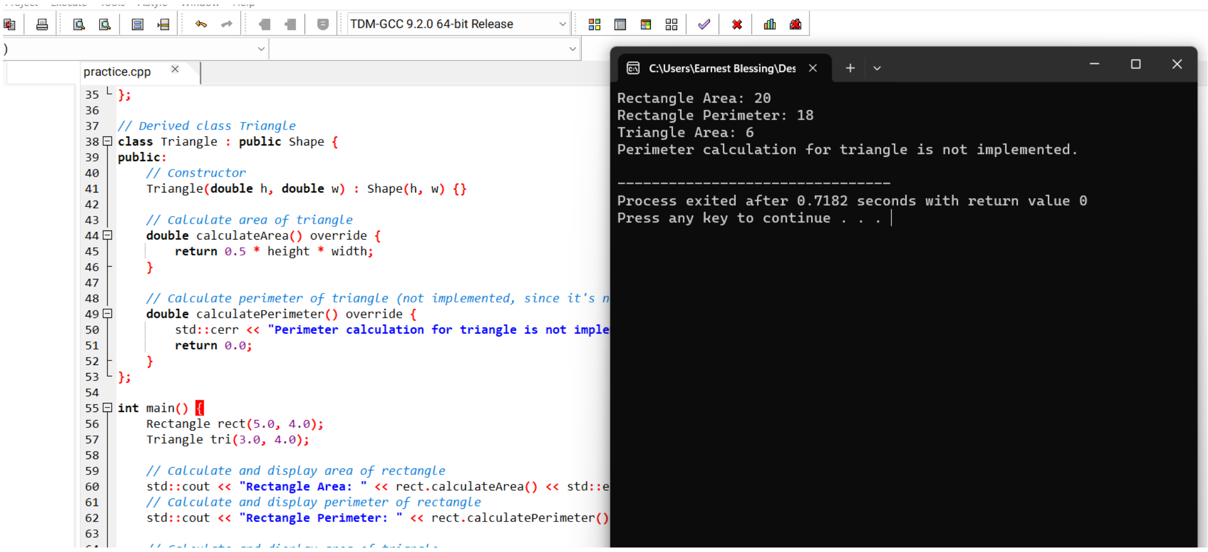
7. Develop a program how to use parameterized constructor to initialize the data member of class with user defined values of library management process getting book detail in stack, user detail,check the availablity of book in stock, distribute the book to the user



```cpp
        if (available) {
            available = false;
            std::cout << "Book with ID " << id << " has been borrowed." << std::end
        } else {
            std::cout << "Sorry, the book with ID " << id << " is not available for
        }
    }

    // Method to return the book
    void returnBook() {
        if (!available) {
            available = true;
            std::cout << "Book with ID " << id << " has been returned." << std::end
        } else {
            std::cout << "Invalid operation. The book with ID " << id << " is alrea
        }
    }
};

int main() {
    std::stack<Book> library;

    // Add some books to the library
    library.push(Book("The Great Gatsby", "F. Scott Fitzgerald", 101));
    library.push(Book("To Kill a Mockingbird", "Harper Lee", 102));
    library.push(Book("1984", "George Orwell", 103));

    // Display available books
    std::cout << "Available Books:" << std::endl;
    while (!library.empty()) {
```

```
Available Books:
Book ID: 103, Title: 1984, Author: George Or
well, Status: Available
Book ID: 102, Title: To Kill a Mockingbird,
Author: Harper Lee, Status: Available
Book ID: 101, Title: The Great Gatsby, Autho
r: F. Scott Fitzgerald, Status: Available

--------------------------------
Process exited after 0.7988 seconds with ret
urn value 0
Press any key to continue . . .
```

8.Create a base class called shape with data members for height and width derive two class rectangle and triangle for the base class write number function to calculate the area and perimeter of each class



```cpp
};

// Derived class Triangle
class Triangle : public Shape {
public:
    // Constructor
    Triangle(double h, double w) : Shape(h, w) {}

    // Calculate area of triangle
    double calculateArea() override {
        return 0.5 * height * width;
    }

    // Calculate perimeter of triangle (not implemented, since it's n
    double calculatePerimeter() override {
        std::cerr << "Perimeter calculation for triangle is not imple
        return 0.0;
    }
};

int main() {
    Rectangle rect(5.0, 4.0);
    Triangle tri(3.0, 4.0);

    // Calculate and display area of rectangle
    std::cout << "Rectangle Area: " << rect.calculateArea() << std::e
    // Calculate and display perimeter of rectangle
    std::cout << "Rectangle Perimeter: " << rect.calculatePerimeter()

```

```
Rectangle Area: 20
Rectangle Perimeter: 18
Triangle Area: 6
Perimeter calculation for triangle is not implemented.

--------------------------------
Process exited after 0.7182 seconds with return value 0
Press any key to continue . . .
```