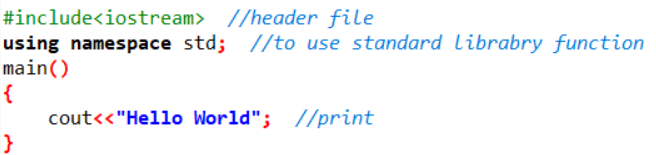
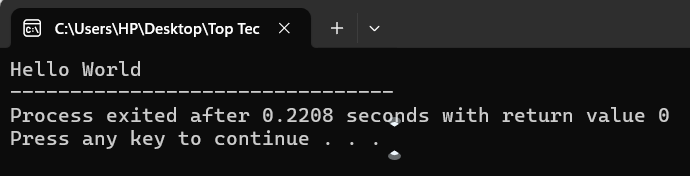
MODULE: 4.1 (C++ Basic)

1. **WAP to print “Hello World” using C++**

****

**Output:**

1. **What is OOP? List OOP concepts.**

The Object-Oriented programming paradigm plays an important role in human computer interface. It has different components that takes real world objects and performs actions on them, making live interactions between man and the machine.

* OOP is an engineering approach to build software system.
* It is a fundamental programming concept, in which object & class are mainly used by developers to connect data & function together.
* OOP uses the concept of object & class to model the real-world entities.
* It provides a clear structure to the program.
* The first step in OOP is to collect all the objects a programmer wants to manipulate & identify, how they relate to each other.
* An object can be defined as a data field that has unique attributes and behavior.
* Consist of a group of co-operating objects.
* Object exchange message to achieve common objective that is called message passing.
* This program implements with the help of object-oriented programming.

**Example of OOP Language:**

Using namespace std; ​

class Hero​

{ ​

// properties​

char name (100);​

int health; ​

char level; ​

} ​

​

**Most of the modern programming language are object-oriented programming language.**

* Java
* C#
* PHP
* C++ & many more.

**Concept of OOP:**

* Class
* Object
* Abstraction
* Inheritance
* Polymorphism
* Encapsulation

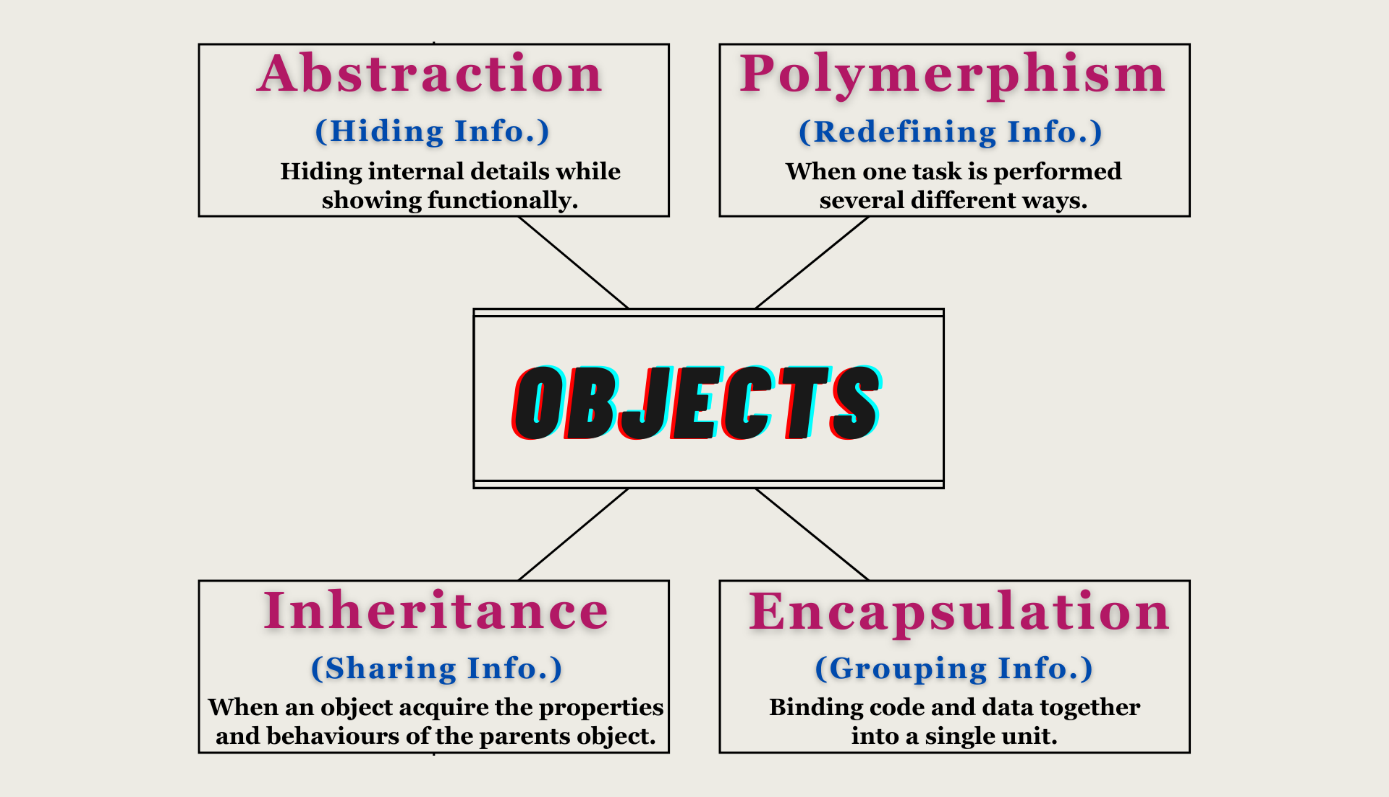
In this Class & Objects are also known as Building Block of OOP Concept.

Abstraction, Inheritance, Polymorphism and Encapsulation are known as Pillar of OOP Concept.

**Building block of OOP Concept:**

* Classes
* Object`
* Method
* Attribute

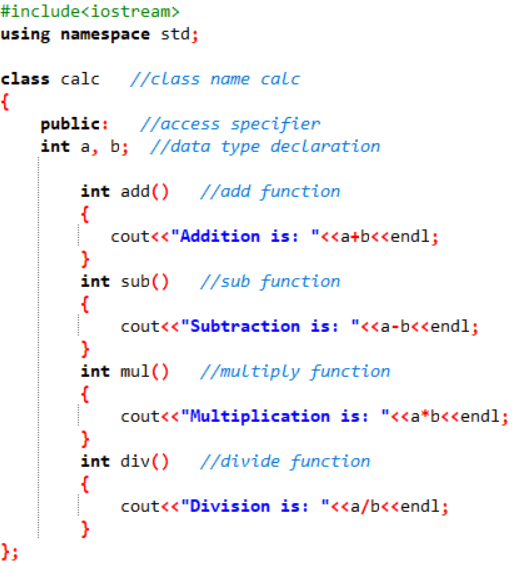
**Structure Of Pillar Of OOP:**

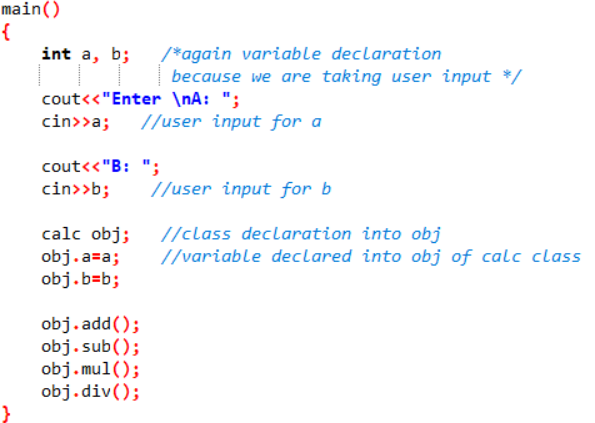


1. **What is the difference between OOP and POP?**

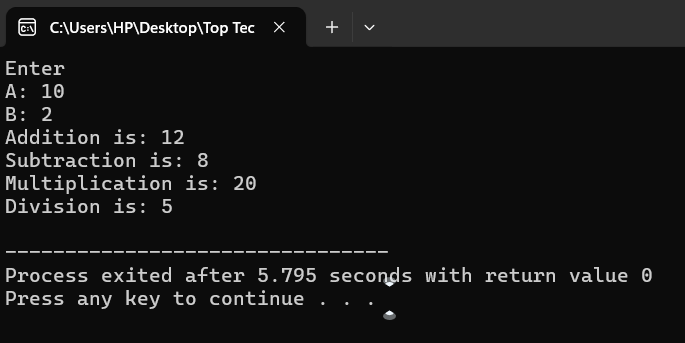
|  |  |  |
| --- | --- | --- |
| **S. No.** | **OOPs** | **POP** |
| **1.** | **It refers to OBJECT ORIENTED PROGRAMMING** | **It refers to PROCEDURE ORIENTED PROGRAMMING** |
| **2.** | **Program is divided into parts is called OBJECT** | **Program is divided into parts is called FUNCTION** |
| **3.** | **It follows BOTTOM-UP approach** | **It follows TOP-DOWN approach** |
| **4.** | **It's easy to manage/ modify** | **It's not easy to manage/ modify** |
| **5.** | **It has more reusability of code** | **It has less reusability of code** |
| **6.** | **It has proper way for hiding data, so it is secure** | **It does not have any proper way for hiding data, so it is less secure** |
| **EX.** | **C++, Java, VB, .NET, etc.** | **C, VB, Fortran, Pascal.** |

**MODULE: 4.2 (Programing with C++)**

1. **WAP to create simple calculator using class .**

****

**Output:**

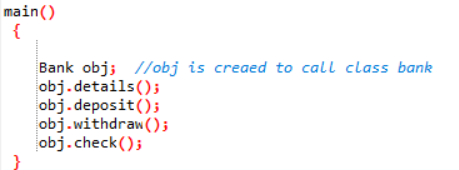
****

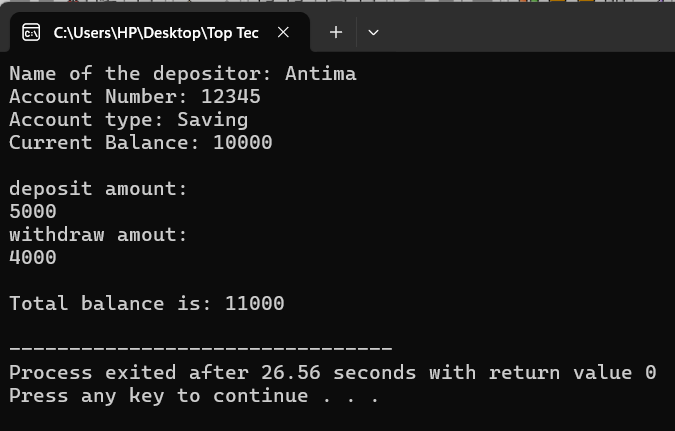
1. **Define a class to represent a bank account.**

**Include the following members:**

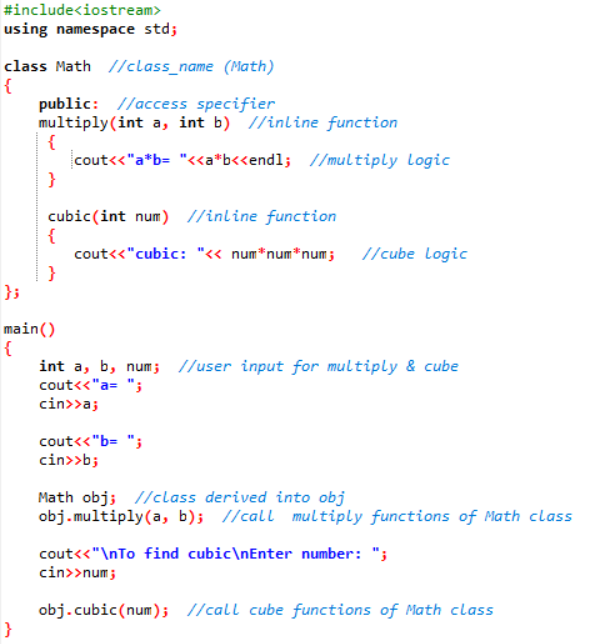
**\*\*3.1) Data Member: -Name of the depositor -Account Number -Type of Account -Balance amount in the account**

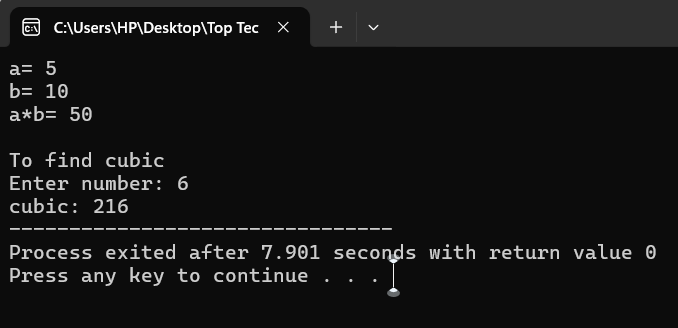
**\*\*3.2) Member Functions -To assign values -To deposited an amount -To withdraw an amount after checking balance -To display name and balance**

**Output:**

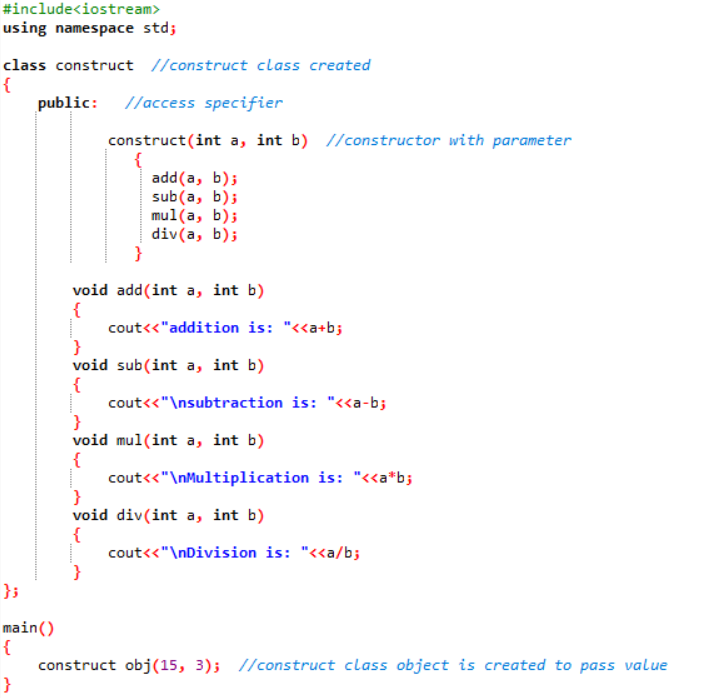
****

1. **Write a program to find the multiplication values and the cubic values using inline function.**

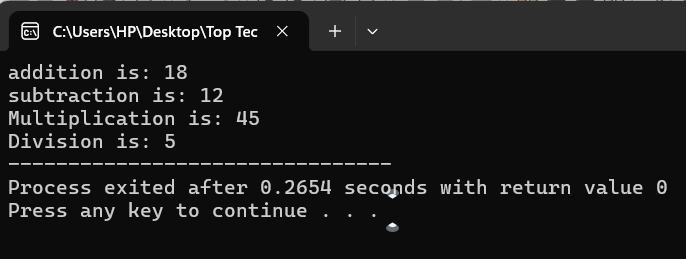
****

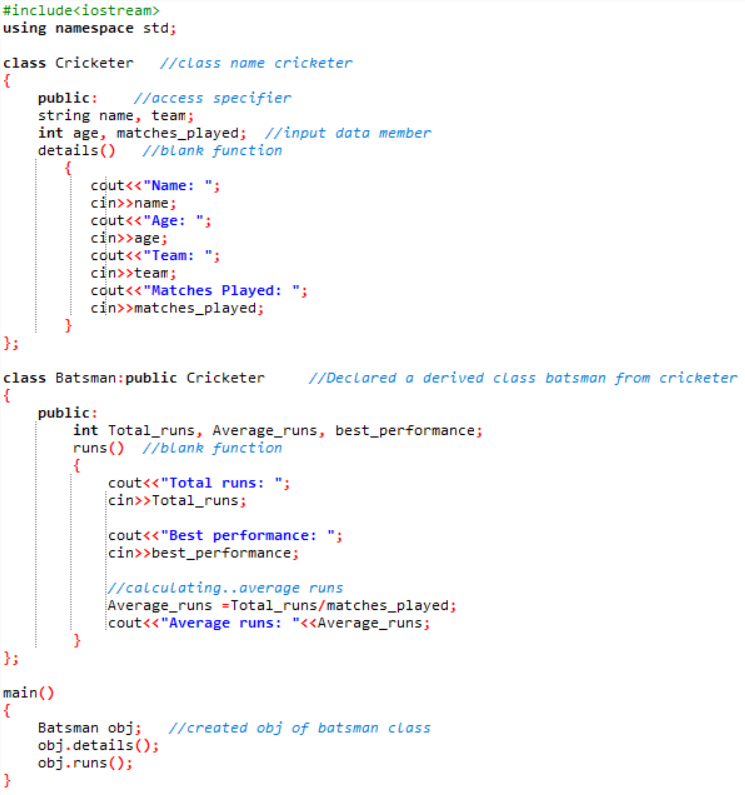
** Output:**

1. **Write a program of Addition, Subtraction, Division, Multiplication using constructor.**

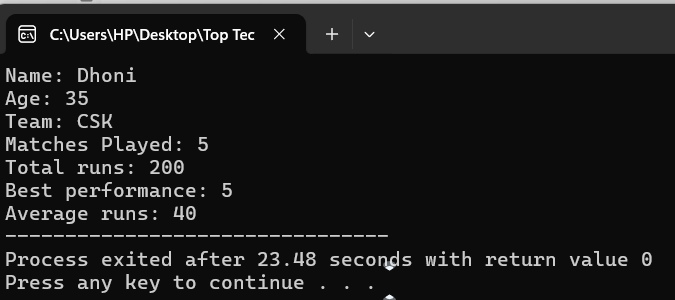
****

**Output:**

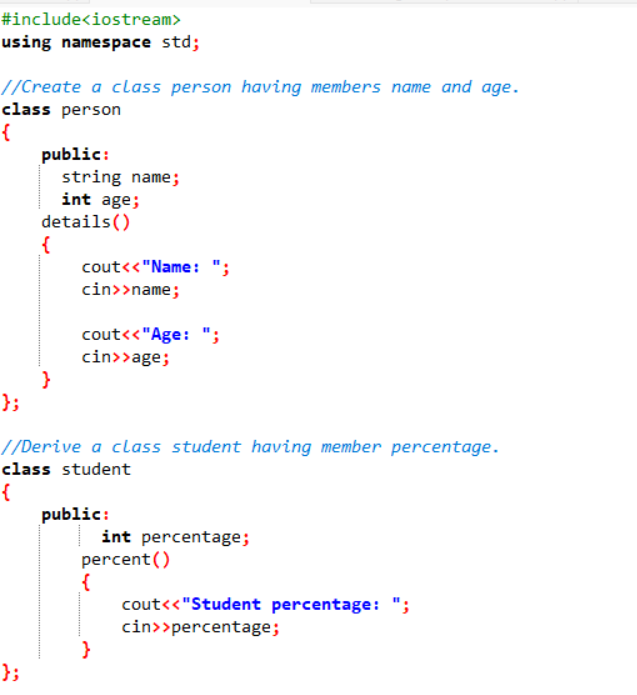
****

1. **Assume a class cricketer is declared. Declare a derived class batsman from cricketer. Data member of batsman. Total runs, Average runs and best performance. Member functions input data, calculate average runs, Display data. (Single Inheritance)**

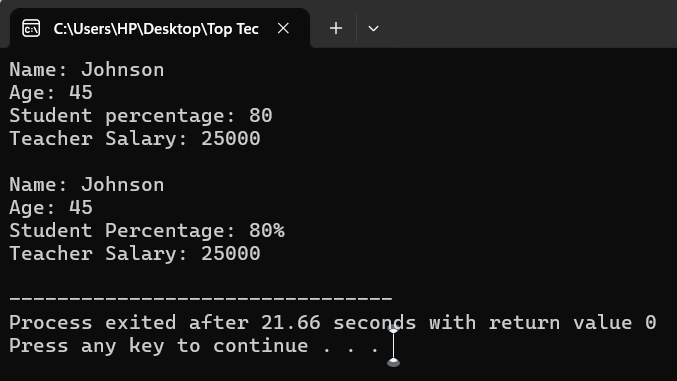
**Output:**

****

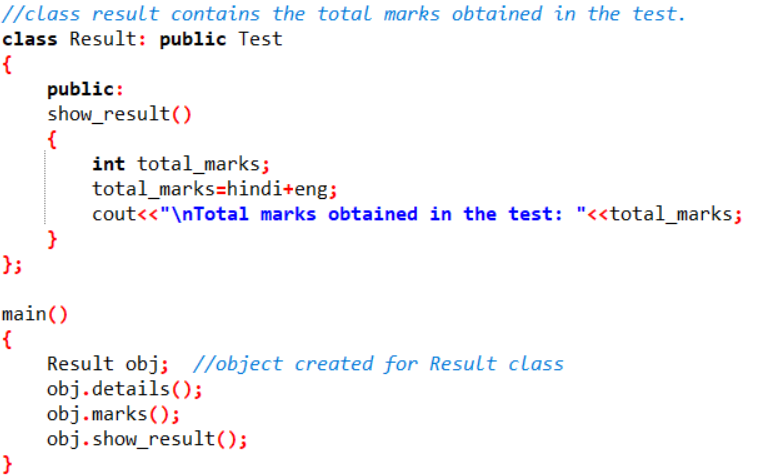
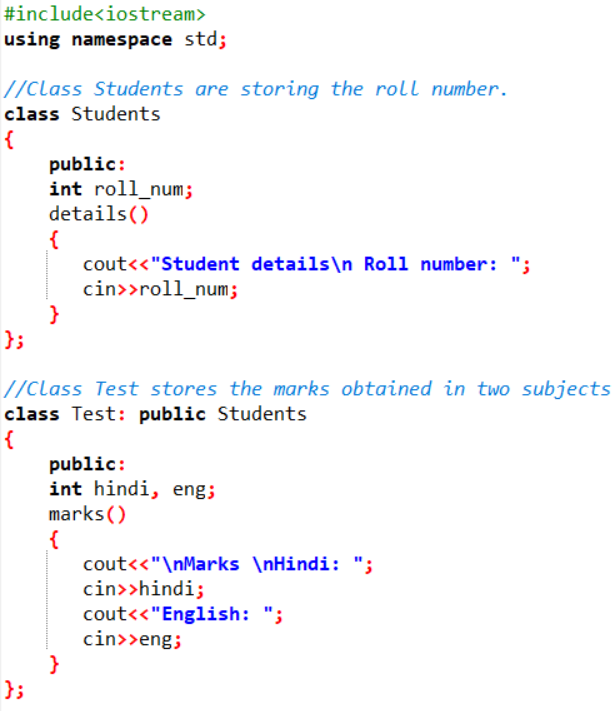
**7) Create a class person having members name and age. Derive a class student having member percentage. Derive another class teacher having member salary. Write necessary member function to initialize, read and write data. Write also Main function (Multiple Inheritance)**



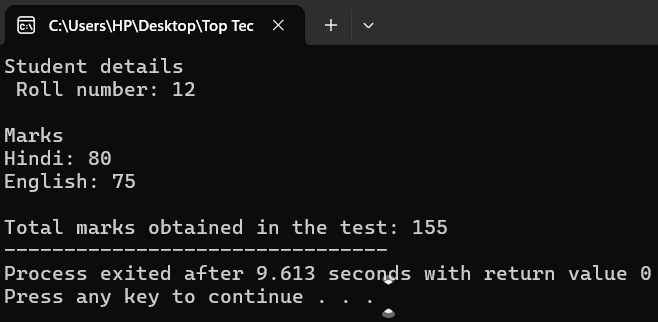
**Output:**

****

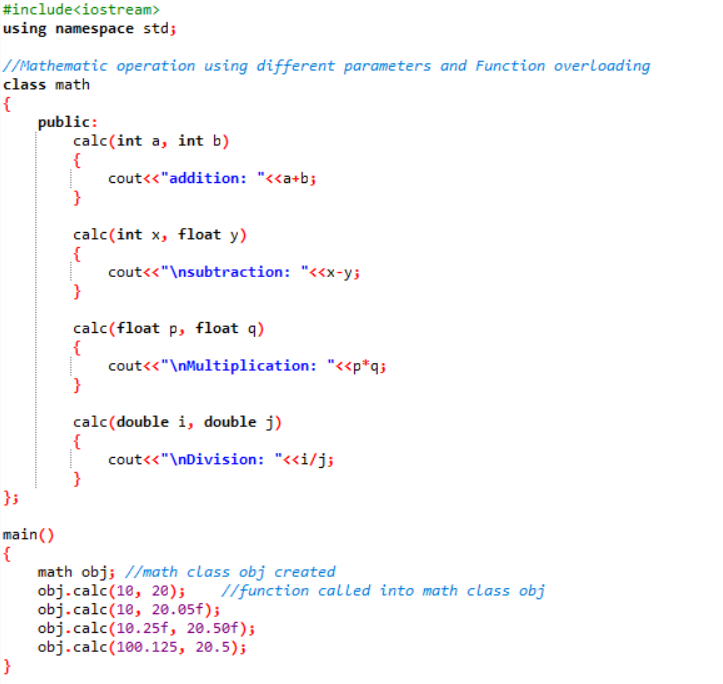
**8) Assume that the test results of a batch of students are stored in three different classes. Class Students are storing the roll number. Class Test stores the marks obtained in two subjects and class result contains the total marks obtained in the test. The class result can inherit the details of the marks obtained in the test and roll number of students. (Multilevel Inheritance)**



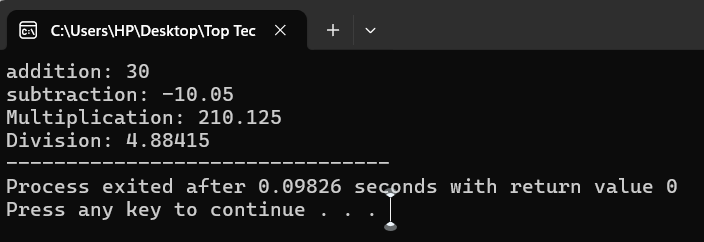
**Output:**

****

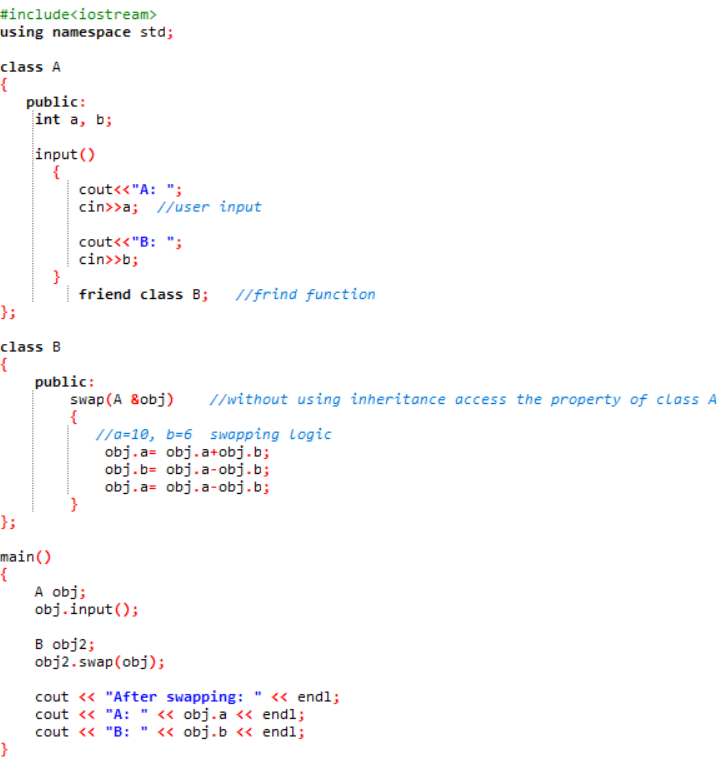
**9) Write a program to Mathematic operation like Addition, Subtraction, Multiplication, Division Of two number using different parameters and Function Overloading.**

****

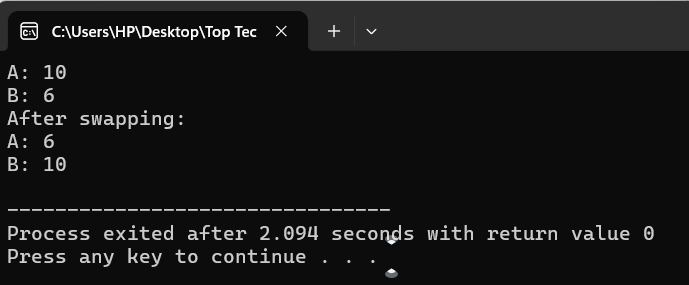
**Output:**

****

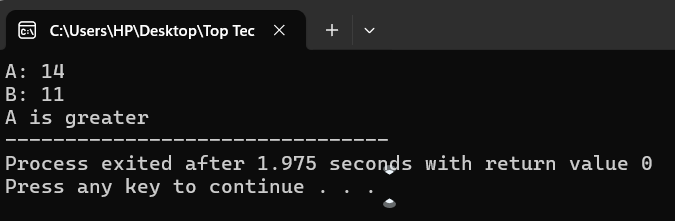
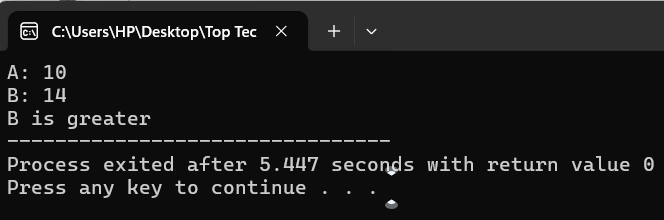
**13) Write a program to swap the two numbers using friend function without using third variable.**

****

**Output:**

****

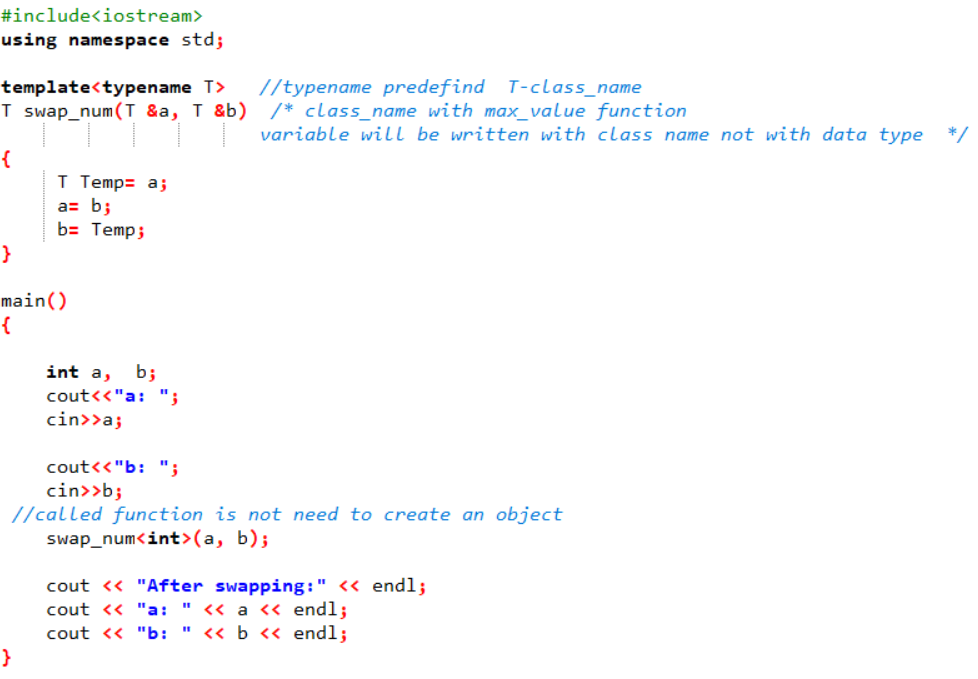
**14) Write a program to find the max number from given two numbers using friend function.**

**Output:**

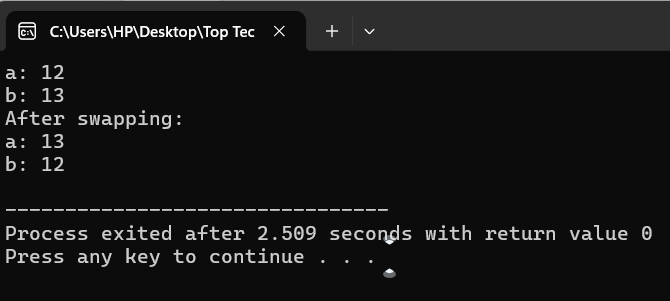
**MODULE: 4.3**

**(C, C++ Templates)**

**15) Write a program of to swap the two values using templates**

****

**Output:**

****