**National University of Computer and Emerging Sciences**



**Lab Manual 07**

**Object Oriented Programming**

|  |  |
| --- | --- |
| Course Instructor | Ms. Syeda Tayyaba Bukhari |
| Lab Instructor (s) | Ms. Fariha Maqbool  Mr. Muhammad Usama Alam |
| Section | BCS-2H |
| Semester | Spring 2023 |

Department of Computer Science

FAST-NU, Lahore, Pakistan

## **Objectives**

After performing this lab, students will practice:

* Friend Functions and Friend Classes

**TASK-1:**

Define a class String. Use overloaded == operator to compare two strings. Make the compare function(==) as friend function

**TASK-2:**

1. Create a **Class A** which has a data member named **int** **number1,** a parameterized constructor to initialize number1.
2. Create a **Class B** which has a data member named **int number2**, a parameterized constructor to initialize number2.
3. Now define a friend function which returns the sum of number1 and number2 from these two classes
4. Call the friend function from main and display the sum

**TASK-3:**

1. Now modify the **Class A** created in Task-2 and declare **Class C** as its friend.
2. **Class C** has a data member **int number3** anda constructor to initialize the number3
3. Create a member function ADD in Class C that adds number1( from class A) to number3(from class C)

You can observe that Class C is able to access the private data members (number1) of its friend class.

1. Call the ADD function of C class from main and display the sum

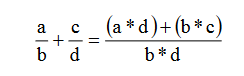
**TASK-3:**

Create a class **RationalNumber** that stores a fraction in its original form (i.e. without finding the equivalent floating pointing result). This class models a fraction by using two data members: an integer for numerator and an integer for denominator. For this class, provide the following functions:

1. A two-argument constructor that initializes the numerator and denominator to the values sent from main. This constructor should prevent a 0 denominator in a fraction, reduce or simplify fractions that are not in reduced form, and avoid negative denominators. If the user skips the values during object creation then it should set the default values of nominator and denominator as 1.

Provide the following operator functions as **non-member friend functions.**

1. An **overloaded operator +** for addition of two rational numbers. Two fractions a/b and c/d are added together as:



1. An **overloaded operator -** for subtraction of two rational numbers.  
   Two fractions a/b and c/d are subtracted from each other as:

