National University of Computer and Emerging Sciences



Lab Manual 08

Object Oriented Programming

|  |  |
| --- | --- |
| Course Instructor | Mr. Bismillah Jan |
| Lab Instructor (s) | Mr. Saif Ali  Mr. Dilawar Shabbir |
| Section | BCS-2E |
| Semester | Spring 2021 |

Department of Computer Science

FAST-NU, Lahore, Pakistan

## Objectives

After performing this lab, students shall be able to perform:

Operator overloading.

* Subtraction as a unary and binary operator.
* Addition and multiplication operations.
* Pre and post increment and decrement operations.
* Stream insertion and extraction operator.

**TASK 1:**

Implement a class called **Fraction**. The **Fraction** class will have two data members:

* int num; // Numerator
* int denum; //Denominator. (Should never be zero)

You need to implement default constructor, overloaded constructor, setter, getters, destructor and overload the operators.

Write a suitable main() for your class to show the functionality of all your functions.

**Your project must have fraction.h, fraction.cpp and main.cpp.**

Implement the following Fraction class and over the described operator.

class fraction

{

int num, denum;

public:

fraction(int = 0, int = 1); //Constructor

void operator!(void) const; // print the fraction

fraction operator-(void) const; // negative of fraction

fraction operator\*(void) const; // reciprocal of fraction

//Fraction’s Arithematic Operators

fraction& operator+=(const fraction&); //frac1 +=frac2

fraction& operator-=(const fraction&); //frac1 -=frac2

fraction& operator\*=(const fraction&); //frac1 \*=frac2

fraction& operator/=(const fraction&); //frac1 /=frac2

// Fraction’s Relational Operators.

bool operator>(const fraction&) const;

bool operator<(const fraction&) const;

bool operator==(const fraction&) const;

bool operator!=(const fraction&) const;

// Overload insertion operator

// Overload extraction operator

class fraction

{

1. Write the code for a copy constructor which copies the values of members of one object of class fraction to another object’s members.

**TASK 2:**

Implement a class named **Month**.

The class should have the following private members:

* **name** A string object that holds the name of a month, such as “January,” “February,” etc.
* **monthNumber** An integer variable that holds the number of the month. For example, January would be 1, February would be 2, etc. Valid values for this variable are 1 through 12.

In addition, provide the following member functions:

• A **default constructor** that sets monthNumber to 1 and name to “January.”

• A **constructor** that accepts the name of the month as an argument. It should set name to the value passed as the argument and set monthNumber to the correct value.

• **Prefix and postfix overloaded ++** operator functions that increment monthNumber and set name to the name of next month. If monthNumber is set to 12 when these functions execute, they should set monthNumber to 1 and name to “January.”

• **Prefix and postfix overloaded − −** operator functions that decrement monthNumber and set name to the name of previous month. If monthNumber is set to 1 when these functions execute, they should set monthNumber to 12 and name to “December.” .