Object Oriented Programming Lab

Lab 08 **Marks 10**

Instructions

Work on this lab individually. You can use your books, notes, handouts etc. but you are not allowed to borrow anything from your peer student. *You are strictly **NOT ALLOWED** to include any additional data-members/functions/constructors* in your class.

Marking Criteria

Show your work to the instructor before leaving the lab to get some or full credit.

What you must do

Program the following task in your C++ compiler and then compile and execute them. *Write the **main** function first and keep testing the functionality of each function once created.*

ADT: Date

Write a class named **Date** having following functionalities

1. The class should have following **three private data members**.
    1. An **integer** named **day** that holds the **day number of the month.**
    2. An **integer** named **month** that holds the **month number of the year**.
    3. An **integer** named **year** that holds the **year number.**

    Value should only be assigned to **day**, if it is in between **1 (default value)** and **31** both inclusive.
    Value should only be assigned to **month**, if it is in between **1 (default value)** and **12** both inclusive.
    Value should only be assigned to **year**, if it is in **greater than or equal to 1900 (default value)**.
2. Provide the implementation of **mutators** for all the data members (day, month and year) of the class.
3. Provide the implementation of **accessors** for all the data members (day, month and year) of the class.
4. Provide the implementation of following **constructors** and a **destructor**
    1. The constructor should accept the **Date's day**, **month** and **year** as arguments. These values should be assigned to the object's appropriate member variables.
    2. The constructor should accept the **Date's month** and **year** as arguments. These values should be assigned to the object's appropriate member variables. The **day** data member should be assigned to the **default value**.
    3. A **default constructor** that initializes all the data members of the class with **default values**.
5. Provide the implementation of following overloaded operators
    1. **stream insertion (<<)** to display the date in the form *November 29, 2021*
    2. **stream extraction (>>)** should prompt the user for a date to be stored in a **Date** object. The operator should ask the user to enter the date in the following format; **dd/mm/yyyy**.
    3. **pre-increment (++)** should increment the **day** data member of the object
    4. **post-increment (++)** should increment the **day** data member of the object
    5. **pre-decrement (--)** should decrement the **day** data member of the object
    6. **post-decrement (--)** should decrement the **day** data member of the object
    7. **subtraction (-) binary** should subtract the one date from another and **return the number of days** between two dates. For example, if **November 10, 2021,** is **subtracted** from **November 18, 2021**, the result will be **8**.
    8. **addition (+) unary** should return **true**, if the date is a **public holiday** (5 Feb, 23 March, 1 May etc. etc.), **false** otherwise.
6. The class should detect the following **conditions** and handle them accordingly:
    1. When a **date** is set to the **last day of the month and incremented**, it should **become the first day** of the following **month**.
    2. When a **date** is set to **December 31 and incremented**, it should become **January 1** of the following **year**.
    3. When a **day** is set to the **first day of the month and decremented**, it should become the **last day** of the **previous month**.
    4. When a **date** is set to **January 1 and decremented**, it should become **December 31** of the **previous year**.
7. Once you have written the class, write **main** function and test its functionality by creating some objects of **Date**.

☺ ☺ ☺ **BEST OF LUCK** ☺ ☺ ☺