CSIS 6033 – Lab 3

Unit Testing

In this lab, you will be writing a suite of unit tests to be run against the *Date* class provided on Blackboard. You should first review the Date.cpp and Date.h to ensure that you understand how the class works. You will need to have the Google Test framework installed and functional before beginning this assignment. Review Week 4 materials for more information.

You are responsible for creating a DateGoogleTest.cpp file that I will then use to run your tests on the *Date* class. Note that for all test cases that you create, they should come back True, i.e. they pass. **All of your unit tests must be done with the Google Test framework**. The details of the unit test suite that you should write are listed below:

- Define five objects as follows:
    - Date x(year,month,day): select any year, month and day that you want
    - Date y(): do not submit any inputs, use the constructor which obtains today's date
    - Date z("YYYYMMDD"): select a date that is one day earlier than x
    - Date d(year,month,day): select any year, month, and day that you want that is different from any previous *Date* variable
    - Define Date::Duration xd_dur from x and d
- Test the boolean operators defined by *Date*:
    - x < y
    - x <= y
    - x != y
    - x == x
    - x >= x
    - x <= x
    - z < x
    - x > z
    - x >= z
    - x != z
- Test the functions defined by *Date*:
    - x.getYear() == "YYYY"
    - x.getMonth() == "MM"
    - x.getDay() == "DD"
    - z.getYear() == "YYYY"
    - z.getMonth() == "MM"
    - z.getDay() == "DD"
    - x.toString() == "YYYYMMDD"
    - z.toString() == "YYYYMMDD"
- Test duration function defined by *Date*:
    - xd_dur.years == "#" (whatever the difference in years is b/w x and d)
    - xd_dur.months == "#" (whatever the difference in months is b/w x and d)
    - xd_dur.days == "#" (whatever the difference in days is b/w x and d)

You should compile and run your test suite to ensure that it works properly. Provide a screenshot along with your code that shows the suite running against the *Date* class and finding no issues.

Graduate students should also answer the following:

- How would you approach testing the private/internal components of the *Date* class?
- Are there any normal or special cases that are not covered in the unit tests listed above? What are they? Be specific.

Submit a zip file to Blackboard which contains your DateGoogleTest.cpp and a word document which includes the screenshot of your test suite running and any other information you think I might need to grade your submission.

For full credit your code should compile, run as described, and be appropriately commented. If I need to know anything in particular about how I should compile your code, include that in your document. Make sure that your code comments include at the top: name, date, semester, course, description of file.

You should submit your solutions to Blackboard by **Thursday, July 22 at 11:59pm**. Your assignment should be of the format CSIS6033_Lab3_pape.zip, where you should substitute your own last name. Be sure to clearly indicate any referenced material that you used to complete this assignment.