COIS 2240H

Winter 2018

Assignment 1

Due Date: 6th of February by 11:59:59PM

Part 1: UML [25 points]

An organization has three categories of employee: professional staff, technical staff and support staff. The organization also has departments and divisions. Each employee belongs to either a department or a division.

Draw a class diagram that captures the above scenario.

Part 2: Java Programming (Recursion/Iteration Comparison) [65 points]

A Fibonacci series of numbers is characterized as the following: 0 1 1 2 3, where the sum of the previous two numbers equal the current number.

With that in mind implement a recursive function called fiboSeriesRec that will produce the Fibonacci series of numbers output up to a predefined n (input by the user). This means that the user will first enter the n^{th} Fibonacci series number and your program will have to calculate the number up until the n^{th} value.

After that implement an iterative function called fiboSeriesIte that will produce the Fibonacci series of numbers output up to a predefined n; iteratively. Recall that iterating typically means using some kind of loop (hint: use a for loop).

Ex:

User input n =3

Fibonacci series program output: 0 1 1

User input n =5

Fibonacci series program output 0 1 1 2 3

Lastly, utilizing a timing library, time the run time to calculate and display the Fibonacci series to the user define n^{th} number. Test a variety of inputs when utilizing a recursive method, and test a variety of different inputs when utilizing an iterative method of generating a Fibonacci series.

This essentially means you will be utilizing a Java timing library, where you initiate some variable like a stopwatch before the numbers are generated to time how long it took and as soon as execution is complete stop timing and output it to the console.

You must run five different tests on both the iterative and recursive program:

n=10

n=20

n= 30

n=40

n = 50

Record the timing for each one and make a graph to compare it.

In 100-200 words analyze which one was faster and why.

Ensure that your code is well commented where it makes sense.

Part 3: Version Control (Git) [10 points]

In order to practice utilizing Git in a collaborative environment, you must first get used to utilizing the functionality when programming on your own. The benefits of version control even when programming individually are immense.

At inception, ensure that you push your program project files onto your remote repository.

> $git add filename.extension (adding project files to the staging area)
>
> $git commit –m"committing the program project files" filename.extension
>
> $git push origin (now your local files which you committed are in your remote repository, you now have a local copy and a backup just in case on your GitHub repository).

Now that you have a remote central repository, create a branch so that you have a stable version of the code to go back to incase anything goes wrong.

> $git branch testing (this creates a branch, which is basically a copy of your code, you may edit the code, test it and if it is stable push it to your remote repository).
>
> $git checkout testing (switching to the testing branch)

If you have completed a significant amount of programming, and you believe you have tested it sufficiently and it is stable you can push to the central repository on GitHub.

> $git push –u origin testing

I expect you to use Git throughout the development process, and to submit a screenshot of your commit logs utilizing the Git log command ($git log).

Deliverables:

1-UML Class Diagram of the scenario (PDF)

2-Program files

3-Screenshot testing verifying that your program actually generates a real Fibonacci series (in a PDF)

4-Screenshot testing verifying that you ran the five different tests with different n values as instructed above (for both the iterative generation and recursive generation of the Fibonacci sequence, this should be in a PDF).

5-Two graphs one for the iterative method and another for the recursive method that plots the times in relation to n (the size of the Fibonacci sequence being generated). Those two graphs should be side by side in a PDF. You may use excel or any other software to generate the graphs.

6-The analysis and comparison of the graphs should be in the same PDF as the actual graphs, the analysis should be brief.

7-Screenshot of commit logs in a PDF.

You must submit all these deliverables in a zip file.

Ensure that you name the PDFs according to this format questionNumber_ lastName_firstName_Assignment1

NOTE: This assignment should be individually completed. Ensure that you do not copy and paste code then claim it as your own. If you utilize a piece of code from a source be sure to cite it within your comments.