CSE 2221 - Project 11

No late submissions will not be accepted. Sunday (4/26/2020) @ 11:59pm.

Task
Gain familiarity with the Java Swing framework and components by using the Model-View-Controller (MVC) design pattern to develop a Java application involving a graphical user interface (GUI).

Original Project Instructions
Project 11 Instructions from CSE2221 Project Site

Program Requirements

- Develop a calculator GUI that does various arithmetic operations on NaturalNumbers
- Aside from computing the square root of very large numbers (in the realm of hundreds of digits), your program should not crash
- If you do have your design deviate from the demo calculator, your solution should be intuitive to use

Tips, Rules, & Things to Note

- Open the "index.html" file in the doc folder of the provided project and read through ALL of the documentation for each of the interfaces
- Many of the method implementations in `NNCalcView` and `NNCalcModel` will be 1-3 lines of code
- Keep the useful constants in your mind (these are class variables defined at the top)
- Avoid copyFrom
- setEnabled (for JButtons) and setText (for JText) are useful methods
- Have the NaturalNumber documentation up while completing this project
- You can also look at the Java documentation for Swing (available online, just search on Google)

Demo Project 11

A demo version of (what should be) the final result of project 11 is available here. Your project doesn't have to exactly match this one although it is heavily recommended. Play around with the demo to get a visual understanding of what is expected of your submission.

Steps

1. Import the project provided at this link
2. If you are struggling with the above step, see the original project instructions for a walkthrough of how to import a project into Eclipse
3. This is already mentioned in the tips section, but once the provided project is imported, go to the `doc` folder and open `index.html` then read through ALL of the documentation for each of the interfaces given to you (specifically the `NNCalcController` is important to read and understand)
4. In the documentation from the last step, read the `NaturalNumberCalculator` section, this describes the expected behavior of the program
5. It is recommended you start with implementing `NNCalcModel1`, then `NNCalcView1`, and last `NNCalcController1`, although you can implement in any order you want

6. Although testing is not required, it is a good idea to unit test your code to ensure correctness, note testing user interfaces can be difficult so proceed with some caution

7. Implement every method is each class to match the method contracts in each of the interfaces

8. Once you have everything implemented and you are convinced your methods are implemented correctly, run the program and do some "system testing" by playing around with the GUI and doing a bunch of various calculations (note: what we have done in the past is considered automation testing, by actually using the GUI we are doing manual testing now, usually manual testing is done first however)

9. Once completed and convinced your code looks good: Zip it up with the naming scheme I recommend and submit to Carmen