CS341 Software Engineering

Homework 1

Write your answers to Part I in an MS Word document and submit to Canvas.
Write your code for Part II in Java. Submit your code and output screens to Canvas.

Part I: Read the Software Engineering chapters taken from Ian Sommerville, provided on Moodle.

1. What are the traditional software phases?
   a. Requirements engineering, design, abstraction, implementation, verification and validation.
   b. Design, optimization, implementation verification and validation and maintenance.
   c. Requirements engineering, design, implementation, verification and validation, and maintenance.

2. Explain why the models listed below are suitable or unsuitable for developing a software control system, such as the control system for the software in an airplane. Which model is the most appropriate for this software system?
   a. Waterfall model
   b. Incremental (Test driven) development model
   c. Prototyping model

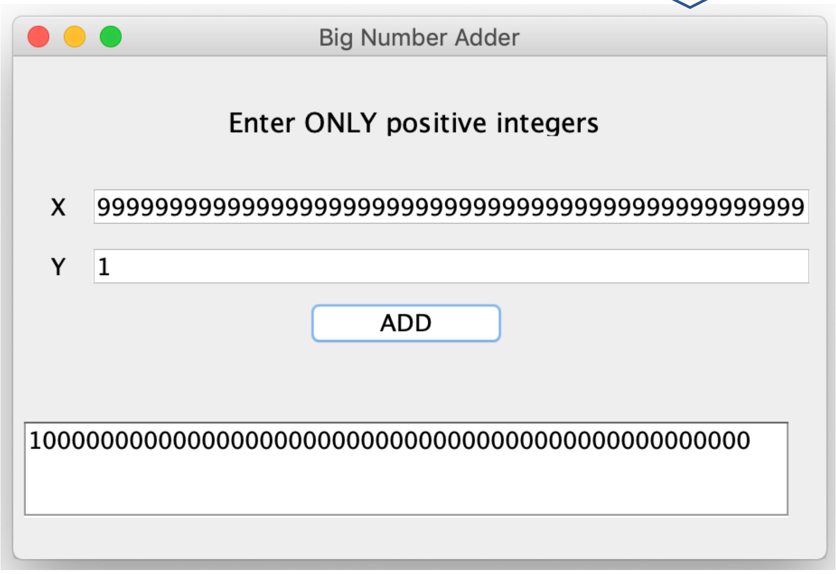
3. Explain why the models listed below are suitable or unsuitable if you expect mid-course corrections. Which model(s) are most suitable for this situation?
   a. Waterfall model
   b. Boehm's Spiral model
   c. Prototyping model

4. What is the difference between software engineering and computer science?
5. Name and describe four areas of ethical professional behavior for software engineers.

Part II: Install WindowBuilder and Checkstyle into Eclipse from the Eclipse MarketPlace.

Practice using both these plugins. If you encounter obstacles during your practice, refer to online discussions, videos, and tutorials. Construct the app shown below.

The App
The largest integer that can be stored in the Java primitive data type **int** is 2147483647. This app should allow the user to enter two positive integers that can be larger than 2147483647. By clicking the ADD button, the user can sum two positive integers of any size.



X: 99999999999999999999999999999999999999999999999999

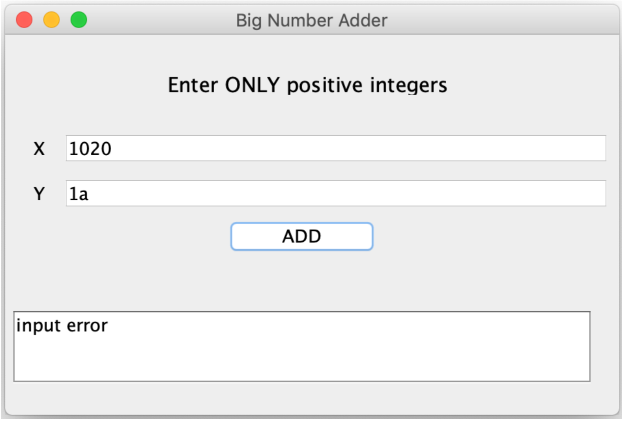
Y: 1

100000000000000000000000000000000000000000000000000

Rule 1: Create a BigNumber class that can be used to model and provide behavior for a big integer.

```java
BigNumber x = new BigNumber(textFieldX.getText());
BigNumber y = new BigNumber(textFieldY.getText());
BigNumber z = x.add(y);
textArea.setText(z.getValue());
```

Rule 2: Include error detection.



X: 1020

Y: 1a

input error