Computing and Algorithms I

Project 2

CS-101 **Fall, 2017**

In project 2 you will be writing a design document for the random guess program. The basic idea is that your program will choose a “random” number, and the user will attempt to guess the number. Specifically, your program will:

1. Ask the user for an upper bound on the random number. For instance, if the user inputs 73, the random number will be chosen from 1 to 73 inclusive. Note that if the user inputs any number smaller than 10, the range will be 1 to 10 inclusive.

2. Generate an int value in the appropriate range (see item 1 above).

3. Ask the user how many guesses he or she wants. The minimum will be one and the maximum will be 4, so if the user enters a 1 or smaller, there is one guess. If the user enters a 4 or larger, there are 4 guesses (unless the user is successful with fewer guesses).

4. The program will ask the user for a guess (if there are any remaining guesses) and will tell the user that the guess was correct, was too high, or was too low. On the last guess, the program will also print a message that the game is over and that the user has won (or lost) if the last guess was correct (or incorrect).

Requirements

You will be writing a design document similar to the one provided to you as part of project 1. In particular, you will be writing a UML class diagram and its corresponding legend. Your design will include at least two methods (main and some other method or methods). Each method will have its own entry in the UML class diagram. Use appropriate software to create the design document (such as a spreadsheet program). The software needs to be able to create a pdf document of your design.

Data Table

Each method in your design will have a data table. Use the format given in the assignment for project 1.

Algorithm

Each method in your design will have an algorithm. Use the format given in the assignment for project 1. Note that each pseudocode statement will correspond to an executable statement in the corresponding program. This was the case with the algorithm in project 1.

Input

Design your solution to use a single Scanner object instantiated from the Scanner class to read input from the keyboard as input by the user.

Processing

Your design will incorporate only these forms of flow of control:

- Sequence
- Invocation
- Selection

In particular, your design will not incorporate any loops (while, do-while, or for statements). Your design will solve the problem of asking the user questions and providing information about the progress of the guessing game.

Output

Design your output to write to the terminal using System.out methods. Clearly indicate the messages your program will display.

Deliverables

On Tuesday October 24 at the beginning of class you will turn in a design document printed from a printer containing the complete design for this project. It will be a print of one or more pdf documents resembling the pdf document supplied to you as part of project 1. You will also submit the pdf and the software file of your design on blackboard. (For example, if you use Excel, you will submit the Excel file as well as the pdf document created from that file.) Your file will have your identification information (name and course) and a title at the beginning of each major part. Look at the pdf document from project 1 for an example.

Grading

The portions of the design will have the following maximum number of points. Full points will be given to complete and correct designs.

1. UML diagram and legend – 10 points
2. data tables for more than 1 method – 5 points
3. algorithms for more than 1 method – 10 points

Turn the printed document in (already stapled) at the beginning of class on Tuesday October 24. Be sure to have the files uploaded to blackboard at this time.