Computing and Algorithms I

Project 3

CS-101 **Fall, 2017**

Project 3 will be an extension of project 2 and will include design and code. There is an updated template for the code included on Blackboard for this project indicating where to place the comments for data tables and algorithms for code with multiple methods. In project 2 you wrote a design document for the random guess program. In project 3, you will update the design and write code to implement your improved design. The changes include that the user may play the game multiple times, and that there is no limit to how many guesses the user may make. Specifically, your program will run a game for the user. At the end of running the game, the program will ask the user if he or she would like to play again. If the user does want to run the game again, the program will do so. If the user does not, the program will end.

Each game will do the following:

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. The program will ask the user for a guess and will tell the user that the guess was correct, was too high, or was too low. If the guess was correct, the user will be told that he or she won and will be told the number of guesses he or she made, at which point this game will end. If the guess was incorrect, the user will be asked if he or she wants to guess again. On a yes response, this guessing process will repeat. On a no response, the user will be told that he or she lost and will be told the number of guesses made, and the game will end. (Note that the program will ask the user if he or she wants to play again at the end of every game).

Requirements

You will be writing a design document similar to the one you wrote for project 2. 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). Furthermore, none of your methods will have more than 40 executable statements. 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. Note the restriction on the number of statements per method (40, as stated above).

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

There will be a loop whose purpose is to play multiple games. A good way to handle this is to have a method in charge of playing one game, and to have that method called in this loop.

Note that there will also be a method to repeat making guesses. As long as the user was incorrect and wants to guess again, this loop should repeat. It is a good idea to have a method to handle one guess, and to call that method repeatedly in this loop.

Output

Design your output to write to the terminal using System.out methods. All communication with the user should be clear.

Deliverables

On Monday October 30 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 in the area for Project3a. (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.

On Tuesday October 31 (Halloween) at the beginning of class you will turn in a printout of your code. Note that all data tables and algorithms from the design will be included as comments in the code. You will also zip your code (using zip) and submit that zip file on blackboard in the area for Project3b. Make sure your code follows the updated java template provided with this assignment.

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

The code will be graded on the following:

1. Program compiles and runs correctly – 15 points.
2. Style for a working program – 5 points.
3. Template for code followed – 5 points