Nassau Community College **Summer I 2016**
CSC 120 CA: Computer Science I **Prof. Christopher R. Merlo**

Project #3: Math Quiz, Part 1
Out: Friday, 3 June 2016
Due: Tuesday, 7 June 2016 11:59:00 pm

Please help NCC conserve resources by not printing this document on campus.

Please note that you are meant to work on this assignment yourself.

1 Objectives

- To practice writing a programmer-defined class
- To practice testing code in a programmer-defined class
- To gain some experience writing some useful code

2 Overview

You have been hired by a school district to write a program for their first grade students who need practice in basic mathematics. The children have learned how to add integers $m + n$, where $0 \leq m \leq 12$ and $0 \leq n \leq 12$, and they have also learned how to subtract integers $i - j$, where $6 \leq i \leq 12$ and $0 \leq j < i$.

In order for this program to be useful, it must be able to generate all possible equations for both operators. For this project, you will write and test a Question class that could be used in the math game.

3 Specifications

3.1 The `Question` Class

You must create a class called `Question`, an object of which represents one question that might be asked in the game. An object of `Question` must store three things:

- The first operand
- The second operand
- Either the + character or the − character

You must also supply the following methods in your `Question` class:

- a *default constructor* that creates a random question, according to the specifications described in the Overview (take a look at the `int nextInt( int n )` method in the `Random` class, http://docs.oracle.com/javase/7/docs/api/java/util/Random.html)

- a `toString` method that generates a string like `"9 - 3 ="` or `"2 + 11 ="` based on what's stored in the object

  Remember: You should test *small amounts of code* at once. Stop now, and write some of the `Project3App` class to test this much of the `Question` class. Once you know this much works, come back and write the next part of the `Question` class and test it. Work back and forth in this way so that you're always coding on a solid foundation.

- a `determineAnswer` method, which calculates and returns, but *does not store*, the answer to the question

3.2 The `Project3App` Class

Creating, displaying, and determining the answer to one `Question` object will not be enough of a test to convince you your code works. Write as little readable code as you need to to create and test fifteen `Question` objects. Your code should generate lines of output like:

```
1: 6 + 3 = 9
2: 8 - 1 = 7
```

etc.

4 Project Report

Create a ***plain-text*** file called `project3.txt` including the statements indicated above, as well as answers to the following questions:

1. How long did it take you to complete this project?
2. What kind of help did you receive on this project, and from whom?
3. What was the most difficult challenge you faced in this project?
4. Are you sure your program works? Why or why not?

Your responses to these questions will be graded based upon grammar and spelling as well as on content. Please use complete sentences. Any file other than a plain-text file will be deleted without ever being read or graded.

5 Submitting Your Project

You must submit your project using the interface on my web site:

http://www.matcmp.ncc.edu/~cmerlo/

No other submissions will be accepted. You may re-submit this project as often as you want; I will only ever see the most recent submission. Notice this means that if you submit before the deadline, and then re-submit after the deadline, your project will be late, and you will lose lateness points. You must submit the following files:

- `Question.java`
- `Project3App.java`
- `project3.txt`

6 Assessment

It is clearly important that your program runs successfully. That is why **30%** of your grade will be based on the **correctness** of your program.

However, it is also important to write code that conforms to the rules we programmers have imposed upon ourselves, to ensure readability. **Ten percent** of your grade will be based, therefore, on your adherence to the **style guidelines** listed on my web site.

Additionally, **proper testing and proper analysis** are the tools we use to convince non-programmers that our code works. Therefore, **ten percent** of your grade will be based on your analysis.

Finally, you will take a short quiz at the end of class on the day of the deadline. The answers to this quiz will account for **50%** of your grade. You will be allowed to refer to a printout of your code during the quiz. This printout ***must*** be of your submission for `Question.java`, and it ***must*** contain your name in the comments. It is ***strongly*** suggested that this printout include line numbers.

7 Things You Should Know

- As with previous projects, you may help each other, but **you are not permitted to share code.** Not even one line. If you're going to talk about this project with each other, leave the conversation empty-handed, and then write the code on your own. Submitting another student's code, or allowing another student to submit yours, will automatically earn both of you a zero on this assignment. Remember that earning a zero due to academic dishonesty also disqualifies you from withdrawing from the class.
- This project is due at 11:59:00 pm on June 7th. Late projects lose 10 points per 24 hour time period or portion thereof, starting at 11:59:01 on June 7th, regardless

of weekends, holidays, weather, computer malfunction, etc. Any submissions uploaded after 11:59:00 pm on Friday, June 10th will be ignored, and a grade of 0 will be recorded.

- Store your data in multiple places. Consider using a system like Dropbox (https://db.tt/KEV2lqS) or SpiderOak (http://www.spideroak.com/). (If you decide to use Dropbox, let me invite you; we both earn an extra 0.25 GB of free space.)