Seneca College **Mar 20, 2019**

Applied Arts & Technology

SCHOOL OF COMPUTER STUDIES

JAC444 **Demo Due date: Mar 29, 2019**

Final Code Submission Date: Mar 29, 2019

Workshop 7

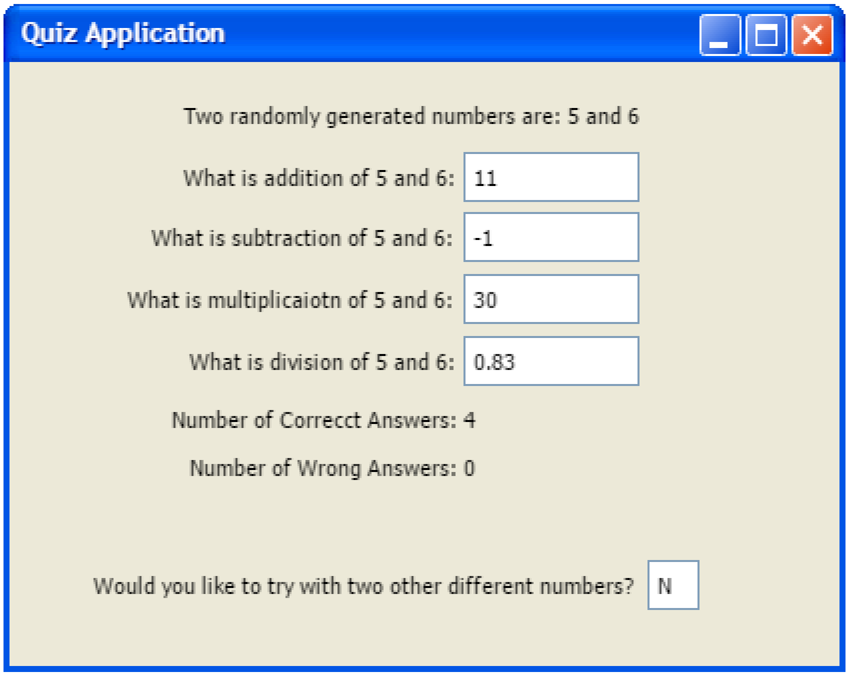
Notes:

- **i.** Each task should be presented during the lab, demo worth 70% of the workshop marks and code uploading worth the other 30%.
- **ii.** Make sure you have all security and check measures in place (with proper use of Exceptional Handling where ever needed), like wrong data types etc.
- **iii.** Make your project in proper hierarchy; introduce proper class coherence in your project. Proper packages and **your project should be handled by only one main method which should be in a TesterClass**.
- **iv.** Given output structure is just for student to have a glimpse what the output can look, students are free to make the output better in any way.

Other inputs can be given during demo, so make sure you test your program properly.

Task 1: (JavaFX required)

Write an application which randomly generate two numbers and asks the user to perform addition, subtraction, multiplication and division on the two numbers. Use “set” to store your answers. Following are some sample screen shots of the program. You can improve the output if you would like, for example you can highlight which questions are not correct as well.



Two randomly generated numbers are: 5 and 6

What is addition of 5 and 6: 11

What is subtraction of 5 and 6: -1

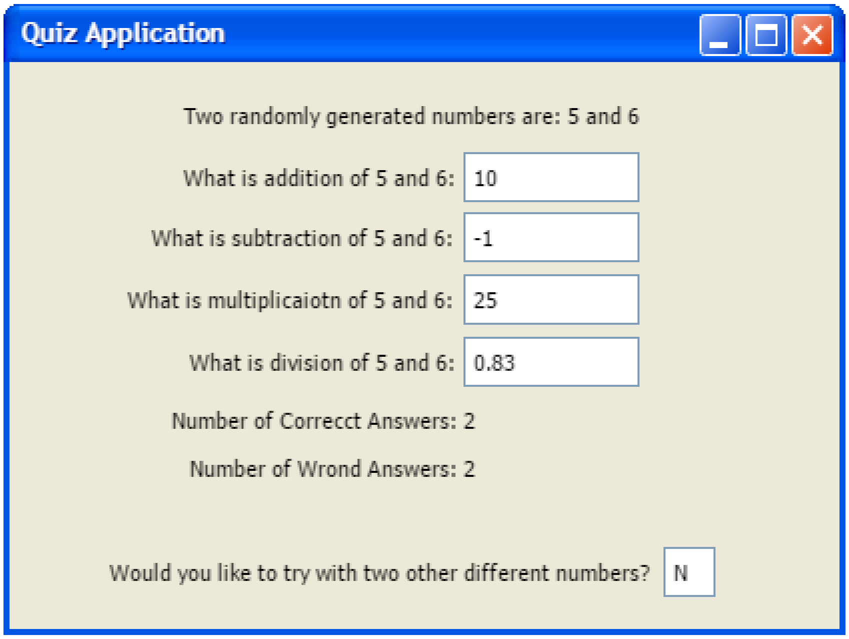
What is multiplicaiotn of 5 and 6: 30

What is division of 5 and 6: 0.83

Number of Correcct Answers: 4

Number of Wrong Answers: 0

Would you like to try with two other different numbers? N



Two randomly generated numbers are: 5 and 6

What is addition of 5 and 6: 10

What is subtraction of 5 and 6: -1

What is multiplicaiotn of 5 and 6: 25

What is division of 5 and 6: 0.83

Number of Correcct Answers: 2

Number of Wrond Answers: 2

Would you like to try with two other different numbers? N

Task 2: (No Java FX)

Describes how to perform matrix addition. Suppose you have multiple processors, so you can speed up the matrix addition. Implement the following method in parallel.

```
public static double[][] parallelAddMatrix(double[][] a, double[][] b)
```

Write a test program that measures the execution time for adding two 2,000 * 2,000 matrices using the parallel method and sequential method, respectively.