CSE 174 – Fall 2019

PROGRAM #5: 32 points – Due Sunday, September 29, by 11:59 p.m.

Outcomes:

- Write programs that obtain user input
- Write programs that use if and else statements
- Write programs that work with random numbers
- Write programs that display numbers formatted according to a given specification
- Format and comment source code that adheres to a given set of formatting guidelines

Scoring:

At a bare minimum, the program you submit must have the assigned source code, and your source code must compile and run without crashing.

- If you do not submit a zip file containing your source code (.java file), your score will be zero.
- If you submit source code that does not compile, your score will be zero.
- If you submit source code that roughly resembles the requirements and it compiles, but it crashes under normal operating conditions (nice input from the user), your score will be reduced by 75%.
- Deductions will be made for not meeting the usual requirements:
  - Source code that is not formatted according to guidelines
  - File and class names that do not meet specifications

	Full credit	**No credit or Partial credit**
Generate random problems (8 points)	You used Math.random() to generate random multiplication and division problems as specified.	Numbers are not random, or do not conform to specifications.
Check user answers (10 points)	You determine whether the user correctly solves the 8 problems.	There are errors in determining whether the user answer is correct.
Display statistical results (10 points)	You correctly compute all the numerical results, including ongoing count, the elapsed time, and the final counts and percentages.	There are errors in your computations, or some computations are missing.
Format console output (4 points)	You formatted output as specified, including displaying problems using the appropriate multiplication and division symbols, and rounding percentages to the correct number of decimal places.	Screen output does not match specifications.

Preliminaries:

- Review Java's `Math.random()` method. Although this method produces `double` values from 0 to almost 1, it is possible to get almost any other range of values you want by using some combination of addition, multiplication, and possibly casting as an `int`. For example, the following will produce random integer values from 1 through 6, inclusive:

```
(int)(1 + 6 * Math.random())
```

- Review Java's `printf()` statement. You may have to research the Internet to determine how to display certain characters, such as a `%` sign, and the multiplication and division symbols.
- Look online to find a way to compute elapsed time in Java. The timer should start when the user sees the first question, and end when the user answers the last question.

Assignment:

Write a program that gives the user 4 random multiplication problems, followed by 4 random division problems to solve. Display the user's ongoing progress, final scores, and elapsed time. Your program should allow the option for easy problems or hard problems, and should match the formatting shown in the sample run as closely as possible.

Requirements:

Your program should meet all of the following requirements:

- The class name should be `MathQuiz`
- All 8 problems should use `int` values.
- Use `Math.random()` for generating your random values.
- The range of random numbers for each problem should be based on whether the user asks for the "easy" or the "hard" options.
    - "easy" means that all numbers, including the answers, will be no more than 100. It should be possible to get values all the way up to and including 100. It should be possible to get numbers as small as 1. Zero should not be possible.
    - "hard" means that all numbers, including the answers, will be no more than 1000. It should be possible to get values all the way up to and including 1000. It should be possible to get numbers as small as 1. Zero should not be possible.
- For division problems, there should never be a remainder. So, 48 divided by 4 should be possible, but 48 divided by 5 should not be possible.
- Display the problems using the multiplication and division symbols shown in the sample run (~~note that the multiplication symbol is *not* the letter x~~, use the letter x as for the multiplication symbol). It is the standard multiplication symbol.
- Display the ongoing scores and final scores exactly as shown above, including rounding percentages to two places after the decimal point.
- Display the total time it took the user to answer all 8 questions.
- Submit your source code file as a zip file.

Sample run:

Your program should match this format as closely as possible. Note that text shown in red is there because the user typed it. You are not supposed to print those.

```
Welcome to my math quiz!
This quiz will give you four multiplication problems,
followed by four division problems.
An easy quiz will include numbers up to 100.
A hard quiz will include numbers up to 1000.
------------------------------------------------
Do you want an easy quiz or a hard quiz?
Enter the word hard or easy: easy

-MULTIPLICATION-------------------------------------
You have chosen easy.
5 x 10 = 50
Correct! 1 answers correct so far.
8 x 7 = 56
Correct! 2 answers correct so far.
8 x 9 = 71
Sorry, 8 × 9 = 72. 2 answers correct so far.
6 x 5 = 30
Correct! 3 answers correct so far.

-DIVISION-------------------------------------------
72 ÷ 9 = 9
No, 72 ÷ 9 = 8. 3 answers correct so far.
10 ÷ 10 = 1
Correct! 4 answers correct so far.
1 ÷ 1 = 1
Correct! 5 answers correct so far.
45 ÷ 5 = 9
Correct! 6 answers correct so far.

-RESULTS--------------------------------------------
You answered the questions in 47 seconds.
Multiplication score: 3 out of 4 (75.00%)
Division score: 3 out of 4 (75.00%)
Overall score: 6 out of 8 (75.00%)
```