

1 Introduction

As it states on the syllabus, projects are to be done on your own and as with all assignments the sharing of files and code is strictly prohibited and constitutes an act of Academic Misconduct. Furthermore, the use of any electronic medium, such as code repositories, forums, blogs, message boards, email, etc. is strictly prohibited and constitutes an act of Academic Misconduct. You are not to discuss the project with anyone other than the professor. You may use the resources that are on the COSC 117 course web page for this class in MyClasses.

As usual, you will submit all your work through the MyClasses page for this class, under Project #1. Make sure you do the formatting Shift+Ctrl+F (or Shift+Command+F on the Mac) before you submit your work. All you need to submit is the Java code file for the program.

The program itself should, of course, be nicely formatted and commented and should follow all the other rules of good programming style. Use the built-in formatting tool in Eclipse and put in some vertical white space to aid in the readability but don't over do it. Variable names should be representative of their purpose. As always, there must be our standard header comment. All variables must have a comment to their use and major blocks of code should contain brief but descriptive comments to their function.

You do not need to write up a formal algorithm for this project but you should think about what needs to be done, what needs to be stored and what needs to be displayed on the screen, before you begin coding. You may also consider doing the project in stages, that is, get some portions working then add in more functionality incrementally.

The grading of the project will take two forms, a sample run and an inspection of the code. If the program does not run you will receive a zero for that portion. So even if the program is not complete you will get a better grade for a partial program that runs verses a program that does not run. So I would suggest a completion in stages approach, as I mentioned above. The run portion of the grading will test the user interface for usability and conforming to the specifications I have outlined above. The code inspection portion of the grade will involve commenting, readability, correct indentation, variable names, structure and style, correctness, and conforming to specifications.

You have a choice of one of two programs to write, either the dice game of Fizzbin or a math quizzing program for elementary students. Do only one of these programs.

2 Fizzbin

The game of Fizzbin is a simple dice game played between two players. In each round of Fizzbin the two players roll two icosahedron (20 sided) dice. Each player can choose to stay with their current roll or reroll either of the two dice but not both. After the reroll a third icosahedron die is rolled by each player and their score for the round is the sum of the three die. The winner of the round is the player with the highest sum with one exception. If a player has a total that is less than half the total of the other player the player with the low roll wins the round. If the roll sums between the two players are equal then the round is a draw and neither player wins that round. The game proceeds until one player wins 10 rounds.

The program must check to see if the user has input valid data. For example, if the user needed to input a Y or N and instead input a U the program should tell the user that the input was invalid and ask for the input again. For the error checking, you may assume that the user inputs the correct data type. So when asked for which die to reroll (an integer) you may assume that they do not input a string that would crash the program. There are ways to test the data type so that even if the user input a string when asked for an integer the program would not crash, we will discuss these later in the course.

A complete run of the game is below. Note that Round 5 has some user input errors and how the program should respond.

```
===== Round 1 =====

First Roll
Player 1 Roll: Die 1: 6      Die 2: 2
Player 2 Roll: Die 1: 4      Die 2: 2

Player 1: Would you like to reroll a die? (Y or N): y
Player 1: Which die would you like to reroll? (1 or 2): 2

Player 2: Would you like to reroll a die? (Y or N): y
Player 2: Which die would you like to reroll? (1 or 2): 2

Final Roll
Player 1 Roll: Die 1: 6      Die 2: 17      Die 3: 15      Total: 38
Player 2 Roll: Die 1: 4      Die 2: 6       Die 3: 9       Total: 19

Player 1 wins this round.

Score: Player 1: 1      Player 2: 0

===== Round 2 =====

First Roll
Player 1 Roll: Die 1: 2      Die 2: 1
Player 2 Roll: Die 1: 19     Die 2: 1

Player 1: Would you like to reroll a die? (Y or N): n

Player 2: Would you like to reroll a die? (Y or N): y
Player 2: Which die would you like to reroll? (1 or 2): 2

Final Roll
Player 1 Roll: Die 1: 2      Die 2: 1      Die 3: 11      Total: 14
Player 2 Roll: Die 1: 19     Die 2: 7      Die 3: 13      Total: 39

Player 1 wins this round.

Score: Player 1: 2      Player 2: 0

===== Round 3 =====

First Roll
Player 1 Roll: Die 1: 7      Die 2: 17
Player 2 Roll: Die 1: 8      Die 2: 8

Player 1: Would you like to reroll a die? (Y or N): y
Player 1: Which die would you like to reroll? (1 or 2): 1

Player 2: Would you like to reroll a die? (Y or N): y
Player 2: Which die would you like to reroll? (1 or 2): 2

Final Roll
Player 1 Roll: Die 1: 19     Die 2: 17     Die 3: 13      Total: 49
Player 2 Roll: Die 1: 8      Die 2: 16     Die 3: 13      Total: 37

Player 1 wins this round.

Score: Player 1: 3      Player 2: 0

===== Round 4 =====

First Roll
Player 1 Roll: Die 1: 5      Die 2: 5
Player 2 Roll: Die 1: 15     Die 2: 19

Player 1: Would you like to reroll a die? (Y or N): y
Player 1: Which die would you like to reroll? (1 or 2): 1
```

```
Player 2: Would you like to reroll a die? (Y or N): n

Final Roll
Player 1 Roll: Die 1: 7      Die 2: 5      Die 3: 16      Total: 28
Player 2 Roll: Die 1: 15     Die 2: 19     Die 3: 9       Total: 43

Player 2 wins this round.

Score: Player 1: 3      Player 2: 1

===== Round 5 =====

First Roll
Player 1 Roll: Die 1: 12     Die 2: 13
Player 2 Roll: Die 1: 16     Die 2: 8

Player 1: Would you like to reroll a die? (Y or N): n

Player 2: Would you like to reroll a die? (Y or N): r
Invalid input please try again.
Player 2: Would you like to reroll a die? (Y or N): y
Player 2: Which die would you like to reroll? (1 or 2): 5
Invalid input please try again.
Player 2: Which die would you like to reroll? (1 or 2): 4
Invalid input please try again.
Player 2: Which die would you like to reroll? (1 or 2): 2

Final Roll
Player 1 Roll: Die 1: 12     Die 2: 13     Die 3: 14      Total: 39
Player 2 Roll: Die 1: 16     Die 2: 1      Die 3: 16      Total: 33

Player 1 wins this round.

Score: Player 1: 4      Player 2: 1

===== Round 6 =====

First Roll
Player 1 Roll: Die 1: 6      Die 2: 8
Player 2 Roll: Die 1: 13     Die 2: 3

Player 1: Would you like to reroll a die? (Y or N): y
Player 1: Which die would you like to reroll? (1 or 2): 1

Player 2: Would you like to reroll a die? (Y or N): y
Player 2: Which die would you like to reroll? (1 or 2): 2

Final Roll
Player 1 Roll: Die 1: 8      Die 2: 8      Die 3: 14      Total: 30
Player 2 Roll: Die 1: 13     Die 2: 18     Die 3: 18      Total: 49

Player 2 wins this round.

Score: Player 1: 4      Player 2: 2

===== Round 7 =====

First Roll
Player 1 Roll: Die 1: 1      Die 2: 19
Player 2 Roll: Die 1: 10     Die 2: 16

Player 1: Would you like to reroll a die? (Y or N): y
Player 1: Which die would you like to reroll? (1 or 2): 1

Player 2: Would you like to reroll a die? (Y or N): n

Final Roll
Player 1 Roll: Die 1: 16     Die 2: 19     Die 3: 17      Total: 52
Player 2 Roll: Die 1: 10     Die 2: 16     Die 3: 15      Total: 41

Player 1 wins this round.
```

```
Score: Player 1:  5      Player 2:  2

===== Round 8 =====

First Roll
Player 1 Roll: Die 1:  4      Die 2: 19
Player 2 Roll: Die 1:  3      Die 2: 16

Player 1: Would you like to reroll a die? (Y or N): y
Player 1: Which die would you like to reroll? (1 or 2): 1

Player 2: Would you like to reroll a die? (Y or N): y
Player 2: Which die would you like to reroll? (1 or 2): 1

Final Roll
Player 1 Roll: Die 1: 16      Die 2: 19      Die 3:  6      Total: 41
Player 2 Roll: Die 1:  6      Die 2: 16      Die 3: 18      Total: 40

Player 1 wins this round.

Score: Player 1:  6      Player 2:  2

===== Round 9 =====

First Roll
Player 1 Roll: Die 1:  8      Die 2:  3
Player 2 Roll: Die 1: 16      Die 2: 14

Player 1: Would you like to reroll a die? (Y or N): y
Player 1: Which die would you like to reroll? (1 or 2): 2

Player 2: Would you like to reroll a die? (Y or N): n

Final Roll
Player 1 Roll: Die 1:  8      Die 2: 18      Die 3: 17      Total: 43
Player 2 Roll: Die 1: 16      Die 2: 14      Die 3:  4      Total: 34

Player 1 wins this round.

Score: Player 1:  7      Player 2:  2

===== Round 10 =====

First Roll
Player 1 Roll: Die 1: 14      Die 2:  4
Player 2 Roll: Die 1:  5      Die 2: 16

Player 1: Would you like to reroll a die? (Y or N): y
Player 1: Which die would you like to reroll? (1 or 2): 2

Player 2: Would you like to reroll a die? (Y or N): y
Player 2: Which die would you like to reroll? (1 or 2): 1

Final Roll
Player 1 Roll: Die 1: 14      Die 2:  5      Die 3:  2      Total: 21
Player 2 Roll: Die 1:  3      Die 2: 16      Die 3: 16      Total: 35

Player 2 wins this round.

Score: Player 1:  7      Player 2:  3

===== Round 11 =====

First Roll
Player 1 Roll: Die 1: 20      Die 2: 18
Player 2 Roll: Die 1: 19      Die 2: 17

Player 1: Would you like to reroll a die? (Y or N): n

Player 2: Would you like to reroll a die? (Y or N): n

Final Roll
```

Player 1 Roll: Die 1: 20 Die 2: 18 Die 3: 2 Total: 40
Player 2 Roll: Die 1: 19 Die 2: 17 Die 3: 6 Total: 42

Player 2 wins this round.

Score: Player 1: 7 Player 2: 4

===== Round 12 =====

First Roll

Player 1 Roll: Die 1: 13 Die 2: 11
Player 2 Roll: Die 1: 13 Die 2: 12

Player 1: Would you like to reroll a die? (Y or N): n

Player 2: Would you like to reroll a die? (Y or N): n

Final Roll

Player 1 Roll: Die 1: 13 Die 2: 11 Die 3: 11 Total: 35
Player 2 Roll: Die 1: 13 Die 2: 12 Die 3: 20 Total: 45

Player 2 wins this round.

Score: Player 1: 7 Player 2: 5

===== Round 13 =====

First Roll

Player 1 Roll: Die 1: 15 Die 2: 7
Player 2 Roll: Die 1: 13 Die 2: 20

Player 1: Would you like to reroll a die? (Y or N): y

Player 1: Which die would you like to reroll? (1 or 2): 2

Player 2: Would you like to reroll a die? (Y or N): n

Final Roll

Player 1 Roll: Die 1: 15 Die 2: 14 Die 3: 1 Total: 30
Player 2 Roll: Die 1: 13 Die 2: 20 Die 3: 17 Total: 50

Player 2 wins this round.

Score: Player 1: 7 Player 2: 6

===== Round 14 =====

First Roll

Player 1 Roll: Die 1: 20 Die 2: 19
Player 2 Roll: Die 1: 5 Die 2: 9

Player 1: Would you like to reroll a die? (Y or N): n

Player 2: Would you like to reroll a die? (Y or N): y

Player 2: Which die would you like to reroll? (1 or 2): 1

Final Roll

Player 1 Roll: Die 1: 20 Die 2: 19 Die 3: 14 Total: 53
Player 2 Roll: Die 1: 17 Die 2: 9 Die 3: 17 Total: 43

Player 1 wins this round.

Score: Player 1: 8 Player 2: 6

===== Round 15 =====

First Roll

Player 1 Roll: Die 1: 7 Die 2: 16
Player 2 Roll: Die 1: 12 Die 2: 2

Player 1: Would you like to reroll a die? (Y or N): y

Player 1: Which die would you like to reroll? (1 or 2): 1

```
Player 2: Would you like to reroll a die? (Y or N): y
Player 2: Which die would you like to reroll? (1 or 2): 2

Final Roll
Player 1 Roll: Die 1: 14      Die 2: 16      Die 3: 20      Total: 50
Player 2 Roll: Die 1: 12      Die 2: 8      Die 3: 4      Total: 24

Player 2 wins this round.

Score: Player 1: 8      Player 2: 7

===== Round 16 =====

First Roll
Player 1 Roll: Die 1: 11      Die 2: 7
Player 2 Roll: Die 1: 15      Die 2: 16

Player 1: Would you like to reroll a die? (Y or N): y
Player 1: Which die would you like to reroll? (1 or 2): 2

Player 2: Would you like to reroll a die? (Y or N): n

Final Roll
Player 1 Roll: Die 1: 11      Die 2: 1      Die 3: 5      Total: 17
Player 2 Roll: Die 1: 15      Die 2: 16      Die 3: 6      Total: 37

Player 1 wins this round.

Score: Player 1: 9      Player 2: 7

===== Round 17 =====

First Roll
Player 1 Roll: Die 1: 3      Die 2: 19
Player 2 Roll: Die 1: 13      Die 2: 6

Player 1: Would you like to reroll a die? (Y or N): y
Player 1: Which die would you like to reroll? (1 or 2): 1

Player 2: Would you like to reroll a die? (Y or N): y
Player 2: Which die would you like to reroll? (1 or 2): 2

Final Roll
Player 1 Roll: Die 1: 4      Die 2: 19      Die 3: 13      Total: 36
Player 2 Roll: Die 1: 13      Die 2: 20      Die 3: 3      Total: 36

This round was a draw.

Score: Player 1: 9      Player 2: 7

===== Round 18 =====

First Roll
Player 1 Roll: Die 1: 20      Die 2: 11
Player 2 Roll: Die 1: 9      Die 2: 8

Player 1: Would you like to reroll a die? (Y or N): n

Player 2: Would you like to reroll a die? (Y or N): y
Player 2: Which die would you like to reroll? (1 or 2): 2

Final Roll
Player 1 Roll: Die 1: 20      Die 2: 11      Die 3: 10      Total: 41
Player 2 Roll: Die 1: 9      Die 2: 11      Die 3: 4      Total: 24

Player 1 wins this round.

===== Game Over =====

Final Score: Player 1: 10      Player 2: 7

Player 1 wins the game.
```

3 An Arithmetic Quiz Program

Write a program that will administer a ten-question arithmetic quiz to the user. There can be addition, subtraction, multiplication, and division problems. The questions should be appropriate for elementary school students who are just beginning to learn arithmetic. All the numbers will be integers. The number of digits for addition problems will be at most two, although the answer could have three. For multiplication problems, at least one of the numbers is to be a one digit number. For subtraction, the answer cannot be a negative number. For division, the answer must be an exact integer, so that a problem like $\frac{28}{5}$ would not be possible but one like $\frac{28}{7}$ would be fine.

For each of the ten problems, the computer will pick the kind of problem — addition, subtraction, multiplication, or division — at random. The numbers in the problem are to be chosen at random as well. So, for an addition problem $A + B$, the A and B will be chosen at random. Present the problem to the user and get the user's answer. Compute the correct answer and check the user's response. Give the user two chances to get each problem correct. If they get it right on the first try, they get full credit (10 points). If they get it right on the second try, they get half credit (5 points). If they don't get the correct answer in two tries, tell them the correct answer, and don't give them any points. At the end of the quiz, tell them how many points they got, out of a possible 100.

The program is to have input and output that looks nice and makes it easy for the user to tell what is going on. It must be friendly to the user (This is for kids!). Specifically,

- For addition, the question layout must look like the examples below. The columns of the numbers are to line up and be right justified. The question mark should always be below the units column.

```

    58
  + 91
  ----
    ?

```

Input your answer:

```

    26
  +  9
  ----
    ?

```

Input your answer:

- For subtraction, the question layout must look like the examples below. The columns of the numbers are to line up and be right justified. The question mark should always be below the units column. Since we are not allowing negative answers for this, the larger number is to be placed on top.

```

    70
  - 25
  ----
    ?

```

Input your answer:

```

    70
  -  5

```

```

-----
      ?

```

Input your answer:

- For multiplication, the question layout must look like the examples below. The columns of the numbers art to line up and be right justified. The question mark must always be below the units column. If one of the numbers is 2 digits in length and the other is one digit in length (remember, at least one of them must be a single digit) put the two digit number on top.

```

      7
    X 6
-----
      ?

```

Input your answer:

```

     17
    X 6
-----
      ?

```

Input your answer:

- For division, the question layout must look like the example below. Use fraction notation, with an equal sign and question mark to the right and vertically centered with the fraction. Remember that the answers are to be integers here, so the numerator must be evenly divisible by the denominator.

```

    54
---- = ?
     2

```

Input your answer:

A portion of a sample run is below. The <<< Removed for Handout >>> simply means that I removed the program output for these questions.

```

    70
   - 25
-----
      ?

```

Input your answer: 45
Correct!

```

    58
   + 91
-----
      ?

```

Input your answer: 139
Your answer is incorrect, try again.
Input your answer: 149
Correct!

```

      7
    X 6

```

?

Input your answer: 42
Correct!

49
X 4

?

Input your answer: 166
Your answer is incorrect, try again.
Input your answer: 186
Both of your answers were incorrect, the correct answer is 196

54
---- = ?
2

Input your answer: 27
Correct!

<<< Removed for Handout >>>

Your final score is 75 out of 100.