

# COMP1618 Exercise 4 – Loops

## Introduction: DiceSimulation.java

This is a simulation of rolling dice. Actual results approach theory only when the sample size is large. So we will need to repeat rolling the dice a large number of times (we will use 10,000). The theoretical probability of rolling doubles of a specific number is 1 out of 36 or approximately 278 out of 10,000 times that you roll the pair of dice. Since this is a simulation, the numbers will vary a little each time you run it.

We will continue to use control structures that we have already learned, while exploring control structures used for repetition. We shall also continue our work with algorithms, translating a given algorithm to java in order to complete our program. We will start with a while loop, then use the same program, changing the while loop to a do-while loop, and then a for loop.

## Task 1: While loop

1. Use the files *Dice.java* and *DiceSimulation.java*. Make sure to place them both in the same project and you can compile both programs. ***Dice.java* is complete** and will not be modified in this lab, but ***DiceSimulation.java* is incomplete**. Since there is a large part of the program missing, the output will be incorrect if you run *DiceSimulation.java*.
2. You will be modifying the *DiceSimulation* class **only**. I have declared all the variables. You need to add what the method does. Convert the algorithm below to Java and place it in the main method after the variable declarations, but before the output statements. You will be using several control structures: a **while** loop and an **if-else-if** statement nested inside another **if** statement. Use the indenting of the algorithm to help you decide what is included in the loop, what is included in the **if** statement, and what is included in the nested **if-else-if** statement.

*Repeat while the number of dice rolls are less than the number of times the dice should be rolled.*

*Roll the first die*

*Get the value of the first die*

*Roll the second die*

*Get the value of the second die*

*If the value of the first die is the same as the value of the second die*

*If value of first die is 1*

*Increment the number of times snake eyes were rolled*

*Else if value of the first die is 2*

*Increment the number of times twos were rolled*

*Else if value of the first die is 3*

*Increment the number of times threes were rolled*

*Else if value of the first die is 4*

*Increment the number of times fours were rolled*

*Else if value of the first die is 5*

*Increment the number of times fives were rolled*

*Else if value of the first die is 6*

*Increment the number of times sixes were rolled*

*Increment the number of times the dice were rolled*

3. Compile and run. You should get numbers that are somewhat close to 278 for each of the different pairs of doubles. Run it several times. You should get different results than the first time, but again it should be somewhat close to 278.

### Task 2: Using Other Types of Loops

1. Change the while loop to a do-while loop. Compile and run. You should get the same results.
2. Change the do loop to a for loop. Compile and run. You should get the same results.

**Congratulations! You completed the exercise on Java Decisions.**