Exercise 1: Simulating multiplication
In early computers, there were no multiplication operations in the instruction set. Instead, multiplication of integers was accomplished through successive addition in software.

a. How would you calculate a product using successive addition? Write a flow chart to show how to calculate 4 * 5.

b. Now think about the more general problem of calculating K*N where K and N are both integers. Alter the original flow chart so that it solves this more abstract problem.

c. What is the Java code that corresponds to this flow chart?

Exercise 2: Mystery loop
Given the following mystery loop in Java, hand simulate it and figure out what it does. Assume n is an integer.

do {
    if ((n%2) == 0) { System.out.println(0); }
    else { System.out.println(1); };
    n = n / 2;
} while (n != 0)

a. A good idea for determining the function of this code is to try it out on some inputs for n. So, hand simulate this code for n = 5 and n = 8. Can you tell what it computes?

b. What would be good test cases for this code? Why?
Exercise 3: Prime numbers
a. We want to write an algorithm that given an integer \( n \) can tell if \( n \) is prime.
Think of a solution procedure that can answer this question.

b. Flowchart your algorithm as a group.

c. Code your algorithm in Java. You may need to use some methods from the Java libraries.

Exercise 4: Making drawings with loops
This exercise involves using loops to draw different shapes.

a. Describe a nested loop that will draw a square on output by printing \( m \) lines of \( m \) stars (*) on the page.
   o Draw a flow chart of your nested loop.
   o Code your loop in Java and run it to see if it ‘works’.
   o How would the algorithm change if you wanted to draw a ‘hollow’ square?
   o How would the algorithm change if you wanted to draw a rectangle that is \( m \) by \( k \) instead of a square?

b. What if you wanted to draw a triangle? How would you have to change the program for a rectangle to draw a triangle?

c. Now consider drawing a circle using a loop. How would you do this task? (Hint: you may need to use your knowledge of geometry and some specialized functions in the Java library.)

d. Using our drawing program methods, code your algorithm from part c in Java. Try your program on a sample radius of size 50 and observe the results on the screen.