Exercise 8

Looping constructs

By the end of this exercise you will be able to

- Use Java's `while`, `do` and `for` looping constructs.

Introduction

The following is a brief outline of the Java's three looping constructs: `while`, `do` and `for`. If you fetch the file `LoopTest.java` now, you will see three methods named: `powerOf2A`, `powerOf2B` and `powerOf2C`. Each of the three methods does the same job, computing powers of two, but each uses a different looping construct.

In `powerOf2A`, the `while` statement is the most straightforward, the test condition `counter != 0` is evaluated on entry into the loop, and then every time through the loop until the test fails and the loop stops.

The `do` loop in `powerOf2B` is identical in execution to the `while` loop except that the test condition is not evaluated on entry to the loop, only after the first run through the loop and so on after that.

The `for` loop in `powerOf2C` is identical in execution to the `while` loop. If you look closely, you will see that the `for` loop has fewer lines than the `while` loop and this is because certain lines have been moved into the top of the `for` loop. The designers of the `for` loop would have you believe that this economy is more than just a matter of saving space, it is also a matter of producing tidier code, because all mention of the `counter` variable that is used for controlling the loop is at one line at the start of the loop. With the `while` loop, this code is in several places and therefore, the designers argue, more difficult to understand.

Questions

1. Fetch the file `LoopTest.java`.

2. In the `doStuff` method, some lines of code to test out the methods `powerOf2A`, `powerOf2B` and `powerOf2C`.

3. There is a bug in the `powerOf2B` method because it does not behave identically to the other two methods in the case when `n` is zero. Put an `if` statement at the top of this method to make it handle the case of zero properly.

4. By copying the pattern of `powerOf2A`, `powerOf2B` and `powerOf2C`, write methods `printLineA` and `printLineB` that work identically to the method `printLineC`, except that they use `while` loops and `do` loops. Add some code to the `doStuff` method to test them out.