

Dates of Importance:

- Tuesday, September 19: (Optional) Suggested Problems Due for Quiz Extra Credit
 Thursday, September 21: (Required) In-class Exercise
 Tuesday, September 26: (Required) In-class Quiz

Objectives:

This exercise will get you (further) experience with:

- declaring variables of various types.
- using Java objects.
- assignment statements.

Preliminaries

Before attempting to complete this assignment, you should be comfortable with the material from Lab 1 and Chapters 1, 2, and 4.3 of the Horstmann text book. As a result, you should be comfortable with all of the material from Lab 1, all of the preliminary problems from Lab 1. and you should be able to complete the following problems before solving the actual lab problem.

1. *Chapter 2:* p. 70–71: problems R2.1, 2.5, R2.8 (you *MUST* choose methods from the Java API!), R2.9, R2.12–R2.15 (*read the problems carefully*), R2.16–R2.19; p. 72–73: problems E2.2, E2.7 (you may assume that no String contains the '@' character, *but make sure that your code would work on any String value ... not just "Hello, World!"*), E2.8, E2.12
2. Re-write your solution to E2.7 so that all of your calls to the `replace` method are on the same line in your Java program *and* you only have (at most) one semicolon on that line.
3. *Chapter 4:* p. 166–167: problem E4.6. Note that 1 mile is (roughly) 1609.34 meters; there are 5280 feet in 1 mile; 1 foot is 12 inches. For clarity, your program should give the number of miles in the specified number of meters, the *total* number of feet in the specified number of meters, and the *total* number of inches in the specified number of meters - noting that the number of inches should be larger than the number of feet, and the number of feet should be larger than the number of miles.

NOTE: You may submit ***personally hand-written*** solutions to ***all*** of the above problems ***at the beginning*** of the class period before the lab. This can earn you up to 2 bonus points on the quiz to be given during the next class meeting after the lab session, but only if you answer ***all*** of the problems!

The Exercise

Write a program that:

1. Prompts for and reads a large “integer” from the keyboard, which may contain any number of digits (numbers like 9876543210987654321098765432109876543210 are just fine). Note that these numbers are too big for a Java `int` or `long` variable, but could be thought of as a String. *Hint:* You might say that such numbers are **mathematically Big Integers**.
2. Determines whether or not the number is likely to be prime, with a likelihood of $1 - \frac{1}{2^{100}}$. As a result prints *one* of the following two messages:

It is likely `true` that the number is prime.

or

It is likely `false` that the number is prime.

Example Runs

The following represents a few sample runs, separated by horizontal lines with output in plain text and *user input in italics*.

What is your large integer? *9876543210987654321098765432109876543210*
It is likely false that the number is prime.

What is your large integer? *32416190071*
It is likely true that the number is prime.

What is your large integer? *17*
It is likely true that the number is prime.

What is your large integer? *27*
It is likely false that the number is prime.

****PRE**-lab Work**

When you come into class for the lab exercise, you must have a corresponding program that:

1. compiles without any errors and ...
2. is a sincere attempt at a solution to the problem. Dr. Blythe's judgment on this will be final, no exceptions.

If you do not meet *both* of the above requirements, you will be given a 0 for this lab and will have to leave class immediately - *no exceptions!!!*

Preparation Notes

- You are *required* to work on this *before* the actual lab. If you do not, you will receive a zero for the lab and will be required to leave class for the day immediately.
- You will be given a minor change to this assignment to complete as part of the in-class exercise. This change will be so minor that if you have done the preparation yourself, it will likely take you less than ten minutes to complete; of course if you don't do your own work in preparing for the lab, the modification may be difficult to complete by the end of class (and completion by the end of class is a *requirement*).
- WARNING: For some people this lab will be a 15-30 minute exercise (if that). For others, it may be a 40+ hour exercise. *Do not wait until the night before this lab is due (when you no longer have 40+ hours) to start this lab and discover you fit in the latter group!!!!*
- If you find yourself trying to write a loop, an if, statement, (recursive) functions, or anything similar to solve this lab, then you have made the lab *millions* of times harder than it should be. Consider looking things up in the Java API instead!
- If you find this lab challenging, be prepared to re-double your efforts; the concepts in this lab are fundamental to the remainder of the semester!