Parameter Problem Set

This handout is meant to help you check that you understand some of the important concepts that do not always come across in the assignments. The problem sets are pencil-and-paper exercises rather than programming problems, but feel free to use the computer to check your work. Problems of this sort, however, may appear on an exam, so the goal is for you to be able to solve them without electronic assistance.

1. True/False questions

For each of the following statements below, indicate whether it is true or false in Java:

1a) The value of a local variable named `i` has no direct relationship with that of a variable named `i` in its caller. _____

1b) The value of a parameter named `x` has no direct relationship with that of a variable named `x` in its caller. _____

1c) Assigning a new value to a parameter changes the value of the corresponding variable in the caller's argument list. _____

1d) Local variables of type `int` are automatically initialized to 0 when a method is called. _____

1e) The `return` statement allows methods to return more than one value. _____

1f) Predicate methods always return a value of the primitive type `boolean`. _____

2. Short answer

Beginning students of programming may find it hard to understand the relationships among variables in the different methods that make up a program. The instructor may hear that it would be easier if a particular variable name, such as `total` or `i`, always referred to the same conceptual entity (whose value could change) throughout the program in which that variable appeared. In your own words, describe what would be wrong with that approach?

Answer to question 2

3. Short answer

Suppose that you wanted to explain the concept of arguments and parameters to a sibling (or, more likely, a parent) with no background in programming at all. What everyday, real-world example would you use to make it clear why an algorithmic procedure (or any generalized notion of a set of steps for carrying out a particular task, to be carried out by a person or by a machine) might want to include some form of parameterization?

Answer to question 3

4. Short answer

In a few sentences, describe the difference between local variables and instance variables. Make sure that your answer includes any differences in (a) how those variables are declared and (b) how long the values of those variables persist.

Answer to question 4

5. Tracing method execution

For each of the two following programs, show the output produced when it runs.

5a)

```
/*
 * File: Mystery.java
 * ------------------
 * This program doesn't do anything useful and exists only to test
 * your understanding of method calls and parameter passing.
 */

import acm.program.*;

public class Mystery extends ConsoleProgram {

   public void run() {
      ghost(13);
   }

   private void ghost(int x) {
      int y = 0;
      for (int i = 1; i < x; i *= 2) {
         y = witch(y, skeleton(x, i));
      }
      println("ghost: x = " + x + ", y = " + y);
   }

   private int witch(int x, int y) {
      x = 10 * x + y;
      println("witch: x = " + x + ", y = " + y);
      return x;
   }

   private int skeleton(int x, int y) {
      return x / y % 2;
   }
}
```

Answer to question 5a

5b)

```
/*
 * File: Hogwarts.java
 * -------------------
 * This program doesn't do anything useful and exists only to test
 * your understanding of method calls and parameter passing.
 */

import acm.program.*;

public class Hogwarts extends ConsoleProgram {

   public void run() {
      bludger(2001);
   }

   private void bludger(int y) {
      int x = y / 1000;
      int z = (x + y);
      x = quaffle(z, y);
      println("bludger: x = " + x + ", y = " + y + ", z = " + z);
   }

   private int quaffle(int x, int y) {
      int z = snitch(x + y, y);
      y /= z;
      println("quaffle: x = " + x + ", y = " + y + ", z = " + z);
      return z;
   }

   private int snitch(int x, int y) {
      y = x / (x % 10);
      println("snitch: x = " + x + ", y = " + y);
      return y;
   }
}
```

Answer to question 5b