

# Programming Design, Spring 2013

## Suggested Solution for Homework 02

Instructor: Ling-Chieh Kung  
Department of Information Management  
National Taiwan University

### Problem 1

Suppose we want to calculate  $x^y$ , where  $y$  is an integer. The calculation is easy: If  $y$  is zero, the result is zero. Otherwise, just multiply  $x$  by itself for  $y - 1$  times, and then find its reciprocal if  $y$  is negative. However, if  $y$  is a fractional number, we will need a complicated way of doing the calculation. Though I do not know how this is implemented, it is conceivable that this should be much more complicated and take much more time. It is then natural to offer two implementations of the `pow()` function, one for each type of the exponent: We need a general implementation to deal with the general fractional exponent case, but if the exponent is an integer, we want to invoke the simpler implementation specifically for integers to save some time.

### Problem 2

For each of the following statements, indicate the sequence of executing all operations. For example, for the statement

```
cout << 3 + 8 * 5;
```

the answer should be “ $8 * 5$  goes first, and then  $3 + 40$ , and finally `cout << 43`.”

- (a) First  $3 + 8$ , then  $11 * 5$ , then `cout << 55`.
- (b) First `cout << "I like to "`, then `cout << "write C++ programs"`.
- (c) First  $-b * c$ , then  $a = -b * c$ , then  $c++$ . Note that it does not matter whether the second or the third step goes first.
- (d) First  $++c$ , then  $b / ++c$ , then  $5 \% 2$ , then  $b / ++c - 1$ , then  $a = b / ++c - 1$ .

### Problem 3

- (a) The program asks the user to input two integers and then calculate the remainder of the division between these two integers. In particular, it is the second integer divides the first integer.
- (b) There is a logical error: If the second integer input by the user is 0, the program will not generate a reasonable result.
- (c) There are at least two ways of fixing the problem. In the first way, before the modulus operation is executed, we should check whether the second integer is 0. If so, we do not execute the modulus operation and print out an error message instead. In the second way, we do this check right after the second integer is input. As long as the validation finds that the second integer is 0, the program should ask the user to enter an integer again.

### Problem 4

The example program can be found at the file “PDSp13\_hw02\_sol.cpp”.