

# Programming Design, Spring 2013

## Lab Exam 1

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

In this exam, there are two problems. You need to write a C++ program for each problem. The grading criteria for the two programs are the same:

- 70% of your grades for this program will be based on the correctness of your output. The online grading system will input 35 sets of testing data and then check your outputs. You may only see the grades of running your program on these data but cannot see the inputs and outputs. These 35 sets count for 70 points, i.e., 2 points for each set.
- 30% of your grades for this program will be based on how you write your program, including the logic and format. Please try to write a robust, efficient, and easy-to-read program.

Please note that your program will be submitted to the online grading system. Therefore, your programs must be able to accept multiple lines of input and stop when the input ends. In short, your programs should be structured in the same way as in the homework.

### Problem 1: weighted average

(50 points) Please write a C++ program that computes weighted averages.

The input contains several lines of numbers. In each line,  $2n + 1$  numbers  $n, x_1, x_2, \dots, x_n, w_1, w_2, \dots, w_n$  will be given. The first number,  $n$ , is the size of two vectors  $x$  and  $w$ . You may assume that  $n \leq 100$ . The first vector  $x$  contains values while the second vector  $w$  contains weights. Your program should calculate the weighted average of  $x$  by using  $w$  as the corresponding weights. All the  $w_i$ s are positive but the sum of  $w_i$ s may be greater than 1. Therefore, the effective weight that should be applied on  $x_i$  is  $\frac{w_i}{W}$ , where  $W = \sum_{i=1}^n w_i$ . With this in mind, your program should output a single number

$$\sum_{i=1}^n \left( \frac{w_i}{W} \right) x_i \quad \text{where } W = \sum_{j=1}^n w_j.$$

For the line of input, numbers are separated by white spaces. For the output, the weighted average should be displays with two digits after the decimal point. Remaining digits should be truncated in any cases. After this number, a new line character/object should be placed.

Below are some examples:

- Input: 2 10 20 0.5 0.5. Output: 15.00.
- Input: 2 10 20 2 8. Output: 18.00.
- Input: 3 5.5555 20.8 40.2 3.4 7.2 9.65. Output: 27.48.
- Input: 6 1 2 3 4 5 6 7 8 9 10 11 12. Output: 3.80.

### Problem 2: counting winning cells in a tic-tac-toe game

(50 points) Please write a C++ program that counts the number of winning cells in a tic-tac-toe game.

The input contains several lines. In each line of the input, you will first be given a positive integer indicating the number of remaining integers in this line. Except the first integer, all the remaining

integers are positive nonrepeating integers between 1 and 9 (both included). Each of these numbers corresponds to a cell in a tic-tac-toe game, with 1 for the left-top cell, 2 for the center-top cell, ..., and 9 for the right-bottom cell. If we exclude the first number, numbers in the odd positions represent the cells occupied by player 1 (she) while those in the even positions are for player 2 (he).

However, not all the nine numbers will appear in a line. In a line, what you may be given is just a partial game which temporarily ends after player 1 occupies a cell. In other words, the numbers given in a line must be one, three, five, or seven. Your program should work for player 2 and tell him how many unoccupied cells are the winning cells of player 1, i.e., once player 1 occupies one of them, she will win. Your program should output a single number that represents the number of player 1's winning cells, followed by a newline object/character. This number may be zero for some partial games.

Below are some examples:

- Input: 5 1 2 3 4 5. Output: 2. The answer is 2 because both cell 7 and 9 are player 1's winning cells. Note that cell 2 is not because it has been occupied.
- Input: 5 5 3 1 4 6. Output: 1. Only cell 9 is player 1's winning cell.
- Input: 7 5 3 1 4 6 9 7. Output: 0. No unoccupied cell is player 1's winning cell.

Please note that the first number in an input line tells you how many cells have been occupied in the partial game.