

# Programming Design, Spring 2014

## Homework 4

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

**Submission.** To submit your work, please upload the following two files to the online grading system at <http://lckung.im.ntu.edu.tw/PD/>.

1. A .pdf for Problems 1 and 2.
2. Your .cpp file(s) for Problems 3 and 4.

Each student must submit her/his individual work. No hard copy. No late submission. The due time of this homework is 8:00am, March 19, 2014.<sup>1</sup> Please answer in either English or Chinese.

### Problem 0

(0 point) Please read Sections 5.1–5.7 and 6.5–6.8 of the textbook.<sup>2</sup> In any case, I strongly suggest you to read the textbook thoroughly before you start to do this homework.

### Problem 1

(30 points) Consider the following program:

```
#include <iostream>
#include <cstring>
using namespace std;

int main ()
{
    char ans[] = "San Francisco";
    char trial[20];

    for (int i = 0; i < 10; i++)
    {
        cout << "Guess my favorite city in US? ";
        cin.getline (trial, 20);

        if (strcmp (ans, trial) == 0)
        {
            cout << "Correct!";
            break;
        }
        else
        {
            cout << "Wrong. " << i << " chances left. \n";
        }
    }

    return 0;
}
```

---

<sup>1</sup>To celebrate the azalea festival, the due date is set on Wednesday, not Monday.

<sup>2</sup>The textbook is *C++ How to Program: Late Objects Version* by Deitel and Deitel, seventh edition.

- (a) (5 points) Try to run this program and identify one logic error. Indicate how to modify the program to fix that bug.
- (b) (10 points) There is a function `strcmp()` used in this program. Search online for its definition and then precisely explain how to use it. In particular, explain how two characters are compared regarding their “value”. Give some examples may help.
- (c) (5 points) How may we use constant variables to improve this program? Explain why.
- (d) (5 points) If we change `<cstring>` to `<string>`, we will get a compilation error. However, if we change it to `<string.h>`, the program will run correctly. Explain why.
- (e) (5 points) What may happen if we change `cin.getline (trial, 20);` to `cin >> trial;`?

## Problem 2

(10 points) Consider the following program, in which the function `printArray()` is the same as the one on p. 20 of the slides used on March 10.

```
#include <iostream>
using namespace std;

void printArray (int [], int);
int main()
{
    int num[5] = {1, 2, 3, 4, 5};
    printArray(num + 1, 4);
    return 0;
}
void printArray (int a[], int len)
{
    for (int i = 0; i < len; i++)
        cout << a[i] << " ";
    cout << endl;
}
```

Note that instead of use `num` as the first argument, here we use `num + 1` when invoking `printArray()`. From the perspective of system memory, explain what is happening here and why we see that outcome.

## Problem 3

(60 points) Given a sentence like “An apple a day keeps the doctors away.”, how many vowels do we have? It is easy to see that there are six ‘a’s, three ‘e’s, no ‘i’s, two ‘o’s, and no ‘u’. In this problem, we will write a program to find the number of vowels for a given sentence.

### Input/output formats

The input will contain 35 lines, each with a sentence composed by English letters (uppercase or lowercase), white spaces, and punctuation marks. The total number of characters in each sentence will be at most 1000. Your program should count the numbers of the five vowels (‘a’, ‘e’, ‘i’, ‘o’, ‘u’), including uppercase and lowercase ones, and print them out in order.<sup>3</sup> In the line of output, the first number should be the total number of ‘a’ and ‘A’, the second number should be the total number of ‘e’ and ‘E’, ..., and the last number should be the total number of ‘u’ and ‘U’. Each output number should be separated by a white space. A new line character should then be appended at the end. For example, suppose a line contains

<sup>3</sup>The functions `tolower()` and `toupper()` defined in `<cctype>` may help.

I am what I am.

as an input line, your program should output

3 0 2 0 0

and then a new line character. As another example, the output for the input line

C++ (pronounced as "see plus plus") is a general purpose programming language.

should be

6 7 2 4 5

and then a new line character.

### Special requirements

You must write a function with the following prototype declaration:

```
void vowelCount (char sentence[], int vCount[]);
```

To invoke this function, a C string must be passed as the first argument. It is the sentence in which the numbers of vowels should be counted. The second argument is a length-five integer array. After the function terminates, `vCount[0]` should store the number of 'a' or 'A' in the C string `sentence`, `vCount[1]` should store the number of 'e' or 'E' in `sentence`, ..., and `vCount[4]` should store the number of 'u' or 'U' in `sentence`.<sup>4</sup> For this problem, the main task will be to implement the function `vowelCount()`. Once you have it, you can write a very short main function.

For this problem, you are not allowed to use C++ `string` (i.e., including the C++ library `<string>` and creating variables whose types are `string`).

### What should be in your source file

For this problem, your `.cpp` source file should contain C++ codes that will both read testing data and complete the above task. You are welcome to use any technique you know. Finally, you should write relevant comments for your codes.

### Grading criteria

The TAs will first open your `.cpp` file to check whether you have declared and implemented the function `vowelCount()`. If no, you will get 0 point. If you have it, your grades will be determined according to the usual rule:

- 70% of your grades for this program will be based on the correctness of your output. PDOGS will compile your program, feed testing data into your program, and check the correctness of your outputs. Each fully correct line of output gives you 2 points.
- 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 if you do not implement the function `vowelCount()`, you will get 0 point regardless the grades PDOGS shows to you.

---

<sup>4</sup>When the array `vCount` is modified in the function body, the argument (the array you pass into `vowelCount()`) will be modified simultaneously. This is, of course, because you are not creating a new array when you invoke `vowelCount()`. You just tell the system where the array is by passing its address (stored in `vCount`) into the function. Therefore, after the function terminates, the caller (e.g., your main function) will have a modified array.

### **(Bonus) Problem 4**

(20 points) For exactly the same input as in Problem 4, we first remove all those characters that are not English characters and then output the longest consecutive sequence of vowels. When multiple sequences all are the longest, output the first one. Below are some examples:

- Input: I am what I am.; output: Ia.
- Input: I know how many students I have in this class.; output: ei.
- Input: I eat apples when I write difficult programming homework.; output: Iea.

Please note that only a new line character should be appended after the longest sequence of vowels (so no period should be printed out). If there is no vowel in the sentence, an empty line should be the output.