

Programming Design, Spring 2015

Homework 1

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

To submit your work, please send an e-mail containing a PDF file for Problems 1, 2, and 4 (optional) and a CPP file for Problem 3 to one of the three TAs. Please name your files as HW01- x , where x is your student ID. For example, if I am submitting this homework, the file names will be HW01-B90705023.pdf and HW01-B90705023.cpp. Using wrong file names will get ***10 points deducted***. Each student must submit her/his individual work. No hard copy. No late submission. The due time of this homework is ***8:00am, March 9, 2014***. Please answer in either English or Chinese.

Before you start, please read Sections 1.7–1.9, 1.14, 2.1–2.8, 3.1–3.5, and 3.7–3.9 in the textbook.

Problem 1

(15 points) Tom wants to print out

Hello World! This is Tom's first program, stored at C:\Users\User\Documents.

by writing the following C++ program:

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

int main()
{
    cout << "Hello World! ";
    cout << "This is Tom's first program, ";
    cout << "stored at C:\Users\User\Documents.";
    return 0;
}
```

However, there are some syntax errors. Please identify the errors and tell him how to correct them.

Problem 2

(25 points) Answer this problem according to the following program. You may assume that the user will always enter two integers no greater than 10^5 .

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

int main(){
int p;int q;cin >> p;cin >> q;
while (p % q != 1){int r = p % q; cout << r << " ";
    p = q; q = r;}
    cout << "\n" << p;
return 0;
}
```

- (a) (10 points) Make the program in a good format. You are graded according to how easy it is to read your program.

- (b) (5 points) Write down the output after entering 96 and 35.
- (c) (10 points) Use your own words to explain what this program does. In particular, please explain the `while` loop in details.

Problem 3

(60 points) In this problem,¹ we write a C++ program to test whether a given positive integer equals m^n , where both m and n are positive integers and $n \geq 2$. For example, $8 = 2^3$ and $243 = 3^5$ fit the requirement; 7 and 242 do not. Once you find that the given number, which is input by the user through `cin`, is m^n for some m and n , you should print m and n out, first m and then n . If a number has multiple ways of being represented as m^n , you should print out the pair with the smallest m . For example, as $64 = 2^6 = 4^3 = 8^2$, you should print out 2 and 6 because $2 < 4 < 8$.

One key feature of your program is that it should accept repeated inputs. The detailed rule is:

- If a positive integer is entered by the user, the program should output m and n following the rules above, or output a single number -1 to show that the number is not in the format of m^n . In the former case, use a single white space to separate m and n . After that, change to a newline and then ask the user to enter another number.
- If zero is entered by the user, the program should terminate immediately.
- In a negative integer is entered by the user, the program should output a warning message, change to a new line, and then wait for the user's next input.

For all messages used as prompts, please design them by yourself so that clear instructions are given to the user.

What should be in your source file

Your `.cpp` source file should contain C++ codes that will complete the above task. You are allowed to use any technique, even those not covered in lectures. Finally, you should write relevant comments for your codes.²

Grading criteria

- 70% of your grades for this program will be based on the correctness of your output. The TAs will compile your program, run your program, and input several testing data to test your program. They will then determine the correctness of your program.
- 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.

Bonus: Problem 4

1. (10 points) Modify the program in Problem 2 so that (1) all variables are initialized and (2) the problem still completes its task when the second entered number is greater than the first one.
2. (10 points) Modify the problem in Problem 2 so that the problem still completes its task when the first entered number can be divided by the second.

¹We will start to use the online grading system to grade your programs starting from Homework 2. This is to help you eliminate the tedious works regarding the formats of input and output so that you may concentrate on writing your first program.

²Read "Formatting a C++ program" in the first-lecture slides for an introduction to comments.