

Programming Design, Spring 2015

Suggested Solution for Homework 1

Solution provider: Shelley Sun

Problem 1

```
#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;
}
```

Problem 2

(a)

```
#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;
}
```

(b)

Output:

26 9 8
9

(c)

The program lets the user input two integers, p and q. The while loop first checks if the condition $(p \bmod q \text{ equals one})$ is true or not. If not, the program defines a variable r which is equal to $p \bmod q$, and prints it out. Then assign $p=q$, $q=r$. It repeats the statements of the while loop until $p \bmod q$ equals one. At last, it prints out the final p.

What this program is trying to do is very similar to Euclidean algorithm (輾轉相除法), which is a method for computing the greatest common divisor (GCD) of two number. However, the final output is not the GCD, if the two number are relatively prime (互質), the output will be the remainder in the last but one iteration of the algorithm. If not, e.g. 21 and 7, the newest q will be changed to 0, and in the next iteration the $p \bmod q$ operation will crash the program. This is because the division by zero has no meaning, and will cause the system crash.

Problem 4

(a)

Modify the variable declarations

```
int p;  
int q;
```

to

```
int p = 0;  
int q = 0;
```

and add

```
if (p < q) {  
    int tmp = q;  
    q = p;  
    p = tmp;  
}
```

before the while loop.

(b)

Modify the condition in the while loop

```
while (p % q != 1)
```

to

```
while (p % q > 1)
```