Computer Programming Language

【Fall, 2018】

Homework 4

Program A: Perfect number (25%)

A positive integer is a perfect number if it is equal to the sum of all of its factors, including one but excluding itself. For example, 6 is a perfect number, since 6 = 1 + 2 + 3, and 1, 2, and 3 are factors of 6. Design a PerfectNumber(long int Num) function that determines whether the supplied number Num is a perfect number. Write a program to find all perfect numbers between 1 and 10000 by calling the function PerfectNumber(long int Num). What is the greatest perfect number you can find?

■ *Web-Cat Submission Check:*

```
int answer1;    // Store the largest perfect number less than 10000 in this global variable
```

Program B: Data processing and sorting (25%)

(a). Create a two-dimensional list of integer part numbers and quantities of each part in stock, and write a function that displays data in the array in decreasing quantity order. No more than 100 different parts are being kept track of. Test your program with the following data:

(b). Modify the function written in part **(a)** to display the data in part number order.

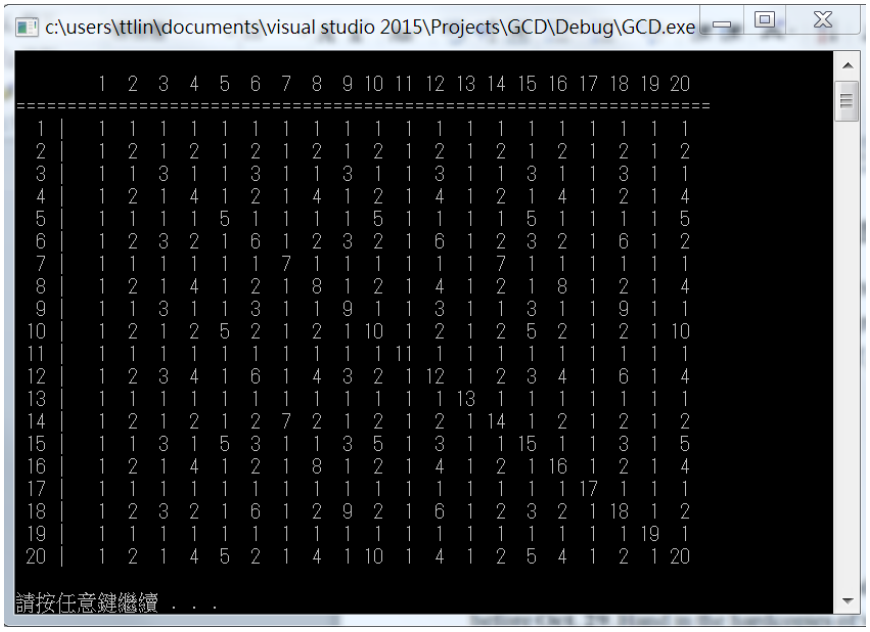
Part No.	Quantity
1001	62
949	85
1050	33
1200	23
867	125
346	59
1025	105

■ *Web-Cat Submission Check:*

```
int answer1;    // Store the part number which is highest in stock quantity
int answer2;    // Store the part number which is lowest in stock quantity
```

Program C：Use of function and nested loops (25%)

The greatest common divisor of integers **x** and **y** is the largest integer that evenly divides both **x** and **y**. Write a function **gcd** that returns the greatest common divisor of **x** and **y**. Also write a C++ program that calls the **gcd** function repetitively to create a table of greatest common divisor of paired integers from 1 to 20, as the following figure shows.



■ *Web-Cat Submission Check:*

```
int answer1;    // Store the greatest common divisor of 10 and 15 in this global variable
int answer2;    // Store the greatest common divisor of 2 and 17 in this global variable
int answer3;    // Store the greatest common divisor of 20 and 20 in this global variable
```

Program D: Dice rolling simulation (25%)

Write a program that simulates the rolling of two dice. The program should use random number generator (C++ rand() function) to roll the first die and should use rand() function again to roll the second die. The sum of two values should then be calculated. Each die can show an integer value from 1 to 6, so the sum of the two values will vary from 2 to 12, with 7 being the most frequent sum and 2 and 12 being the least frequent sums. Your program should roll the two dice 1000, 10000, and 100000 times separately. Use a one-dimensional array to tally the numbers of times each possible sum appears. Display the results in a tabular format. Also, determine if the totals are reasonable by comparing the theoretical probability.

■ *Web-Cat Submission Check:*

```
double answer1;    // Store the probability of rolling 7 (100000 times) in this global variable
double answer2;    // Store the probability of rolling 2 (100000 times) in this global variable
double answer3;    // Store the probability of rolling 12 (100000 times) in this global variable
```

Notes:

1. Please submit your programs (source codes) to the Web-CAT grading system website (http://140.112.94.129:8080/Web-CAT_1.4.0/WebObjects/Web-CAT.woa/) before **Nov. 8**. (3:30PM)

2. Late submission will have a penalty of 10% discount per day of your grade toward a minimum score of 60. No late submission over a week will be accepted.

3. Criteria of grading include: (1) Program functionality; (2). User interface; (3). Structure of the program; (4). Suitable comments; (5). Programming style; (6). Creativity.