Programming (2016/4/12)
Midterm Exam

Student id: _____ **Name:** _____

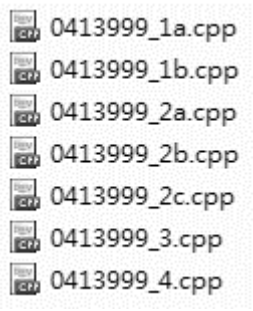
Read me: Please follow the naming rule in this exam otherwise you take the consequences.

In USB folder: TeacherName_StudentName_StudentID

example:

In Answer folder: Please follow each naming rule defined by each question.

example:



1. (a) (20%) Write a program that inputs a five-digit integer, separates the integer into its digits and prints them separately in each new line. [*Hint:* Use the integer division and modulus operators.] For example, if the user types in 96271, the program should print:

```
9
6
2
7
1
```

File name: *Student ID _1a.cpp*

(b) (15%) Write a program that uses a repetition statement to print the following table of values. [Please note the left justified output and two digits to the right of the decimal point]

```
N     N/20    N/5     N*5     N*15
1     0.05    0.20    5       15
2     0.10    0.40    10      30
3     0.15    0.60    15      45
4     0.20    0.80    20      60
5     0.25    1.00    25      75
6     0.30    1.20    30      90
7     0.35    1.40    35      105
8     0.40    1.60    40      120
9     0.45    1.80    45      135
10    0.50    2.00    50      150
```

File name: *Student ID _1b.cpp*

2. Please complete the following:
   (a) (10%) Use `for` loops to print the following pattern:

```
1******
12*****
123****
1234***
12345**
123456*
1234567
```

   File name: *Student ID _2a.cpp*

   (b) (10%) Use the `continue` statement to write a program to display integers from 1 to 100 except multiples of 3 and 7.
   File name: *Student ID _2b.cpp*

   (c) (15%) Repeatedly prompt the user to enter a character to represent four seasons until the user enters EOF. If the user enters one of characters ‘s’, ‘m’, ‘f’, and ‘w’ and its capital letter (大寫), use a `switch` statement to print their respective corresponding seasons (i.e., print ‘Spring’ for ‘s’; ‘Summer’ for ‘m’; ‘Fall’ for ‘f’; and ‘Winter’ for ‘w’); otherwise, print “the entered character is illegal”.
   File name: *Student ID _2c.cpp*

3. (a) (10%) Write a function *draw()* that implements a random integer X ranging from 1 to 4 according to the following probability distribution: `Pr[X=1]=1/10, Pr[X=2]=2/10, Pr[X=3]=3/10, and Pr[X=4]=4/10`, and returns the realized number every time it is called. [*Hint:* You may want to use function *rand()* with seeding in writing this function.]
   (b) (10%) Write a main function that calls *draw()* 100 times, records the frequencies of the drawn results, and display it. For example, you might see a display in the following format at runtime.

```
1: 11
2: 19
3: 32
4: 38
```

   File name: *Student ID _3.cpp*

4. (a) (10%) Write a **recursive function** *gcd*(int, int) that computes the greatest common divider of the two inputted integers where **the base case is when one is the other’s divider with no residual**, and returns such greatest common divider. (An iterative implementation is worth 2 points only.) [*Hint:* The divider and the residual in a division form a sub-problem to the original problem; you can use this to form the recurrence relation.]
   (b) (5%) Write a main function that lets the user input two integers and displays their greatest common divider by using the return value of *gcd*.
   File name: *Student ID _4.cpp*