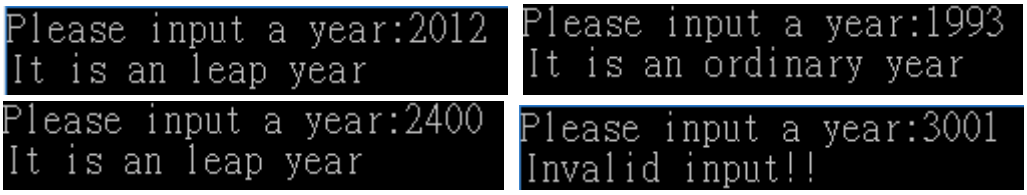
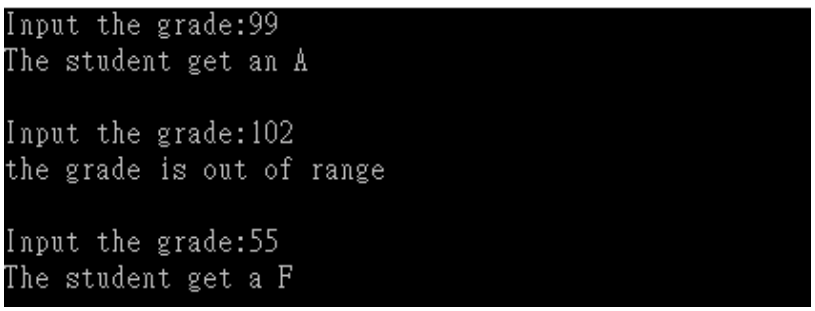
Introduction to Computers and Programming LAB-4 2016/10/12

- ✧ You **cannot** use **array** even if you have learned them.
- ✧ The output must be in our sample output format.
- ✧ If you cannot finish it in time, you should demo your lab work at next lab hours.
- ✧ TAs will update lab records every Monday after the lab hours in the link: http://goo.gl/ZVJu2Y

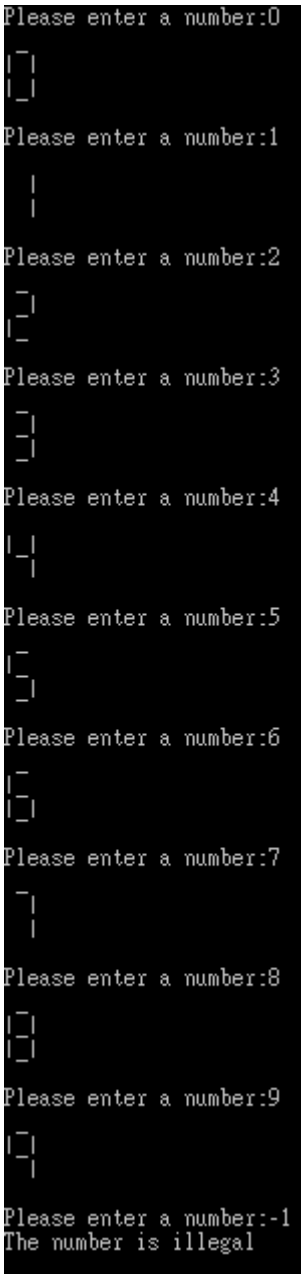
1. There is a leap year every year whose number is **perfectly divisible by four** - **except** for years which are **both divisible by 100** and **not divisible by 400**.
   For example:
   The years 1600 and 2000 are leap years, but the years 1700, 1800, and 1900 are not.
   Please write a program to determine the input year is a leap year or not.
   The input range is $0 < year \leq 3000$ and you need to handle invalid input
   (Show an error message when input is invalid).



2. Write a program that will give a rank for the input grade.
   100~90: A
   89~80: B
   79~70: C
   69~60: D
   59~0: F
   Note: If the grade if greater than 100 or less than 0 you must show that it has an error.



3. Write a program that will show the input number (0~9) in the electric clock format. Besides, if the number is not in the range (0~9), you must print that the number is illegal.



4. *Old McDonald had a farm,*
Ei-I-Ei-I-O
And on this farm he had a chick
Ei-I-Ei-I-O
With a chick chick here
And a chick chick there,
Here a chick, there a chick,
Everywhere a chick chick,
Old McDonald had a farm
Ei-I-Ei-I-OOOO

Old McDonald is a farmer who raises chicken. Today, he picks three chicken randomly and sell them. He wants to record the weight of the chosen chicken, and sort the weighs of them from the lightest to the heaviest one. In addition, he would like to find the lowest, the highest and the average weight. Last, give a rank according to the average weight:

(1) less than (or not equal to) 10 => Poor
(2) between 10 and 29 => Unsatisfied
(3) between 30 and 49 => So-so
(4) between 50 and 99 => Good
(5) More than (or equal to) 100 => Excellent

Now, you are going to write a program to help Old McDonald to finish his job. The inputs are the weights (positive integer) of three chosen chicken. Then show the following information on the program window: **(1) the sorting result from lightest to the heaviest; (2) the weight of the lightest, the heaviest and the average (round down to integer); (3) the rank of the average weight.**

BONUS: You will get extra points if you use switch-case statement to output the rank.

```
Please enter the weight of 1st chicken: 5
Please enter the weight of 2nd chicken: 2
Please enter the weight of 3rd chicken: 1


== After sorting ==
1 2 5

== MIN, MAX, AVG ==
1 5 2

== Ranking ==
Poor
```

```
Please enter the weight of 1st chicken: 100
Please enter the weight of 2nd chicken: 150
Please enter the weight of 3rd chicken: 101


== After sorting ==
100 101 150

== MIN, MAX, AVG ==
100 150 117

== Ranking ==
Excellent
```