考試規定

- 一律採用 Dev-C++ 進行考試，我們會統一以 Dev-C++ 作為批改標準
- 請將所有題目撰寫在同一檔案中，並以 學號.c 或 學號.cpp 的方式命名
- 題目共有七題，單一題目的分數僅有全拿與零分兩種
- 請按照題目給定的格式輸出，若未依格式輸出者一律扣十分
- 第 2、5、6 三題，需使用一個 while loop 讓使用者可以無限輸入，直到輸入 -1 時跳出

考試題目

1. (15%)

Design a program to print your name and student ID

請寫一個程式印出自己的姓名與學號

```
Name: 黃聖瑋
Student ID: M073040035
```

2. (15%)

Write a program that reads in a year as an integer and determines whether it is a leap year or not. (A year is a leap year if it is divisible by 4 but not divisible by 100, or is divisible by 400.) The screen dialog should appear as follows:

寫一個程式去讀取西元年的輸入，並判斷是否為閏年。(閏年定義：若是該年可以被 4 整除但不能被 100 整除，或是可以被 400 整除，即為閏年)。視窗上的輸出範例如下:

```
Enter a year: 2012
Year 2012 is a leap year.
Enter a year: 1900
Year 1900 is a common year.
Enter a year: 2000
Year 2000 is a leap year.
Enter a year: 2014
Year 2014 is a common year.
Enter a year: -1
```

3. (15%)

Write a program that finds the two largest values among five integers. The screen dialog should appear as follows:

寫一個程式讓使用者輸入 5 個數字，並印出最大與次大的值。視窗上的輸出範例如下:

```
Enter five integers:
10 15 21 26 14

Largest is 26
Second largest is 21
```

4. (15%)

Please use *for*, *while* or *do while* to print the times tables. You should align each column with \t. The output should print as follows:

請使用 for, while 或 do while 去印出九九乘法表，且每一列的中間需要用 \t 分隔。視窗上的輸出範例如下:

```
1*1=1   2*1=2   3*1=3   4*1=4   5*1=5   6*1=6   7*1=7   8*1=8   9*1=9
1*2=2   2*2=4   3*2=6   4*2=8   5*2=10  6*2=12  7*2=14 8*2=16  9*2=18
.
.
.
.
.
.
1*9=9   2*9=18  3*9=27  4*9=36  5*9=45  6*9=54  7*9=63  8*9=72  9*9=81
```

5. (10%)

Write a program that reads in a positive integer *n* of at most 8 and an *n*-digit positive integer, then determines whether the *n*-digit integer is a palindrome. The screen dialog should appear as follows:

寫一個程式去讀取一個正整數 n，其值最大為 8。然後再讓使用者輸入一個 n 位數的正整數，並判斷該 n 位數是否為回文。視窗上的輸出範例如下:

```
Enter a positive integer of at most 8: 5
Enter a 5-digit positive integer: 12321
12321 is a palindrome.
Enter a positive integer of at most 8: 6
Enter a 6-digit number: 123456
123456 is not a palindrome.
Enter a positive integer of at most 8: -1
```

6. (10%)

Design a program that input two positive integers, and count the number of carry operations for each of a set of addition problems. For example, 509 plus 104 equals 613 - there is one carry operations(9+4). The length of integers will less than 10 digits.The screen dialog should appear as follows:

寫一個程式來輸入兩個正整數，並計算當這兩個數相加時會有幾個位數發生進位。舉例來說，509+104=613，這邊發生了一次進位(9+4)。使用者輸入的值會少於 10 位數。視窗上的輸出範例如下:

```
123 456
No carry operations
555 555
3 carry operations
999 1
3 carry operations
-1 -1
```

7. (20%)

An *n*-digit number that is the sum of the *n*th powers of its digits is called an Armstrong number. For example, $153 = 1^3 + 5^3 + 3^3$, $1634 = 1^4 + 6^4 + 3^4 + 4^4$, $54748 = 5^5 + 4^5 + 7^5 + 4^5 + 8^5$, so 153, 1634 and 54748 are Armstrong numbers. Write a program that determines whether a number is an Armstrong number. User need to input a positive integer between and including 10 and 10000000, and you need to print all Armstrong numbers between and including 10 and user input (in the range from 10 to 10000000) input by the user. The screen dialog should appear as follows:

一個 n 位數字如果他的每個位數的 n 次方和為該數字本身，我們稱這個數為阿姆斯壯數。舉例來說, $153 = 1^3 + 5^3 + 3^3$, $1634 = 1^4 + 6^4 + 3^4 + 4^4$, $54748 = 5^5 + 4^5 + 7^5 + 4^5 + 8^5$, 所以 153, 1634 與 54748 皆為阿姆斯壯數。請寫一個程式去計算一個數字是否為阿姆斯壯數。使用者需要輸入一個介於 10 到 10000000 間的數，程式必須輸出 10 到 使用者輸入 間所有的阿姆斯壯數。視窗上的輸出範例如下:

```
Enter an integer between and including 10 and 10000000: 20000
Armstrong numbers between and including 10 and 20000:
153
370
371
407
1634
8208
9474
```