

作業 5

Due: 2012.04.13

No later than noon of 2012.04.18 !!

1

1. 一維陣列

- 請寫一程式讓使用者輸入 x 與 y ，爾後宣告兩陣列 $a[20]$ 、 $b[20]$ ，並於 a 陣列中填入以 x 為首項、 y 為公差的等差級數；於 b 陣列中填入以 x 為首、 y 為公比的等比級數。爾後分別將之印出來

- E.g. $x=3, y=2$
- $a: 3, 5, 7, 9, 11, 13, 15, \dots$
- $b: 3, 6, 12, 24, 48, 96, 192, \dots$

- 接著，
- 計算 a 陣列的總和，看看是否與等差級數和的公式相等
- 計算 b 陣列的總和，看看是否與等比級數和的公式相等

$$S_n = \frac{n[2a_1 + (n-1)d]}{2} \quad S_n = \frac{a_1(r^n - 1)}{r - 1}$$

2. 字串操作與一維陣列

- 請寫一程式讓使用者輸入任意字串 (<500)，然後

 - 找出使用者輸入字串的長度
 - 每次將字串中所有字元向右搬一格、並將最尾端的字元搬到字串最開頭後印出，直到字串復原為止。

提示:

- 要找字串的長度，即計算該字串中從頭開始到遇到字元代碼 0 之前共有幾個字元。
- 請不要使用 `strlen` 函式，因為我們還沒教到 ...

執行結果

```
請輸入一個字串: ABCDEFG
字串長度: 7
GABCDEF
FGABCDE
EFGABCD
DEFGABC
CDEFGAB
BCDEFGA
ABCDEFG
```

```
請輸入一個字串: 1234567890ABC
字串長度: 13
C1234567890ABC
BC1234567890A
ABC1234567890
0ABC123456789
90ABC12345678
890ABC1234567
7890ABC123456
67890ABC12345
567890ABC1234
4567890ABC123
34567890ABC12
234567890ABC1
1234567890ABC
```

3. 陣列與迴圈的使用

- 請完成以下的程式，並自行初始化陣列 c 的內容，並讓你完成的程式根據 c 的內容來印出不同數量的米字號 (見下頁範例)

```
int c[10] = { ... };
// 接下來根據 c 的內容來印出如
.
.
.
```

執行結果

```
int c[10]=
{1,4,2,3,7,4,6,0,0,0};
```

```
*
****
**
***
*****
****
*****
```

```
int c[10]=
{1,2,3,4,5,6,7,8,9,10};
```

```
*
**
***
****
*****
*****
*****
*****
*****
*****
```

4. 推土機找土堆

- 請下載 <http://140.118.105.174/Courses/CVB/2012/HW04/> .zip (底線部份以自己學號取代)
- 請利用提供給你的專案，利用 `while` 或是 `do...while` 完成此題
- 目的是讓推土機自己找到土堆，並停留在它的前方。而土堆的位置確定是在棋盤的四週，但位置不固定，且請不要限定棋盤大小(不要去數它)，而是利用 `while` / `do while` 去完成。而推土機的初始位置約略在棋盤正中央。
 - 除了前後移動、轉彎外，會使用到 `while` 或是 `do/while` 或是 `if` 配合以下兩個功能來完成本題
 - `car.IsFrontMoveAble(true)` 用來檢查前方(不限定幾步內)是否有土堆
 - `car.IsFrontMoveAble(false)` 用來檢查前方一格是否有土堆
 - p.s. 如果你試著把土堆搬起來的話，你會發現土堆還會在。它是取之不盡、用之不竭的土堆來源！

7