

# 亞洲大學 資訊工程學系

## 九十九學年 第一學期 期中考

課程：視窗程式設計 (Windows Programming) 2A

授課老師：周永振 助理教授

考試日期：2010/11/12 11:10~14:00

地點：資訊大樓 H113

### 一、選擇填充題 (每題 3 分，共 45 分)

1. 若建立字串資料型別為陣列，在陣列建立時未設初值，則其預設值是 null。
2. 若要使用關鍵字當變數時，必須在關鍵字最前面加上前置字元 @，才能當做程式中的識別項。
3. 宣告變數同時必須給予變數名稱，命名方式則遵循識別項命名規則，同時要設定該變數的 資料型別 以方便在程式進行編譯時配置適當記憶空間存放變數的內容。
4. 若要中途返回迴圈開始處，可在欲返回處插入 continue 陳述式。
5. 運算式就是由運算子和 運算元 組合而成的。
6. 運算時，在運算子前面只需要一個運算元的是為 一元 運算子。
7. 符號常數 是以有意義名稱來取代程式中不會改變的數字或字串。
8. 下列何種資料型態不屬於數值資料? A) char, B) byte, C) long, D) decimal
9. 陣列名稱後面中括號內的整數值稱為註標或 索引。
10. 若建立布林資料型別為陣列，在陣列建立時未設初值，則其預設值是 false。
11. 下列何者不符合識別項命名規則? A) \_pagecount, B) @if, C) B&W, D) Part9
12. 一個方案可建立一個或一個以上專案成為多專案。系統自動將第一個建立專案預設為 起始專案。
13. 陣列的使用是先宣告陣列名稱，再使用 new 關鍵字建立陣列的大小。
14. 依程式執行時該資料是否允許做 四則運算。分成數值資料及 字串 資料。
15. C# 2008 提供一個具有結構且易控制的機制來處理執行時期原程式未考慮的狀況所發生的錯誤稱為 例外處理。

### 二、名詞解釋 (每題 5 分，共 15 分)

1. 何謂“三元運算子”?

運算時需有三個運算子才能做運算，? ... : ... 。語法：  
變數 = 條件式 ? 值 1 : 值 2;  
若條件式結果為 true，則將 值 1 傳給變數，  
否則將 值 2 傳給變數。

2. 何謂“重覆結構”?

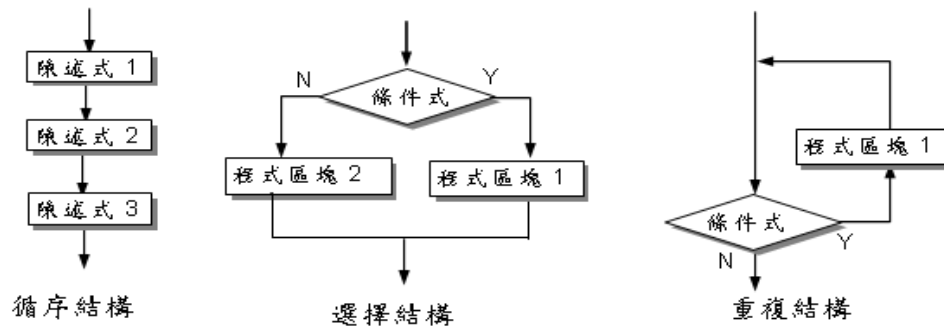
重覆結構 是指設計程式時需要將某部份程式區塊重複執行指定的次數，或 一直執行到不滿足條件為止。

3. 何謂“關鍵字”?

所謂「關鍵字」或稱保留字(Reserve Word)，是對編譯器有特殊意義而預先定義的保留識別項。

### 三、簡答題 (每題 10 分，共 40 分)

1. 請分別畫出循序結構、選擇結構及重複結構之流程圖



2. 令  $x = 6$  且  $y = 2$ ，則執行指令  $x \wedge= y$ ；後  $x$  與  $y$  的值分別是多少？(請寫出運算過程及結果)

$$\begin{array}{rcl} & 0000\ 0110 & \rightarrow x = 6 \\ \text{XOR} & 0000\ 0010 & \rightarrow y = 2 \\ \hline & 0000\ 0100 & \rightarrow x' = 4 \end{array}$$

3. 試說明何謂前測式迴圈與後測式迴圈，並比較其差異

#### 前測式

前測式迴圈 就是將 條件式 放在迴圈的 最前面

依條件式的真假決定是否進入迴圈：

- ① 若滿足條件將迴圈內的程式區塊執行一次，然後再回迴圈最前面條件式。
- ② 不滿足時才離開迴圈

#### 後測式

後測式迴圈 就是將 條件式 放在迴圈的 最後面

第一次不用檢查條件式，直接進入迴圈執行裡面的程式區塊，才判斷條件式的真假：

- ① 若滿足條件會將迴圈內的程式碼執行一次，再檢查位於迴圈最後面的條件式。
- ② 一直到不滿足條件時才離開迴圈

差異：後測式迴圈至少會執行一次

4. 請寫出執行下列程式的執行結果

```

// FileName: series.sln
01 static void Main(string[] args)
02 {
03     int x, sum = 0;
04     Console.WriteLine("\n === 求級數的總和 === \n");
05     Console.WriteLine("  x      2x+1 ");
06     Console.WriteLine(" ===== ");
07     for (x = 1; x <= 5; x++)
08     {
09         Console.WriteLine(" {0}      {1} ", x, 2 * x + 1);
10         sum += 2 * x + 1;
11     }
12     Console.WriteLine(" -----");
13     Console.WriteLine(" 此級數總和為: {0} \n", sum);
14     Console.Read();
15 }

```

```

file:///C:/CSharp/chap03/series/bin/Debug/series.EXE
=== 求級數的總和 ===

  x      2x+1
=====
  1        3
  2        5
  3        7
  4        9
  5       11
-----
此級數總和為 : 35

```