

通識計算機程式設計期中考試題, 4/22/2011

共 8 頁，滿分 100 分

1. 撰寫一或數個 C#敘述達成下列要求: (假設 **using System;** 敘述已經包含於程式中)
 - (a) 宣告 **int** 變數 **k**, **bool** 變數 **b**, **double** 變數 **x** (3%)
 - (b) 在螢幕顯示一行字，要求使用者輸入一個整數 (3%)
 - (c) 自鍵盤讀入一個整數，並將其值存入已宣告之 **int** 變數 **k** (3%)
 - (d) 令已宣告之 **bool** 變數 **b** 等於邏輯敘述：**k** 被 2 除之餘數等於 0 (3%)
 - (e) 若 **b** 為真，在螢幕顯示 "偶數" 否則印出 "奇數" (3%)
2. 撰寫一或數個 C#敘述達成下列要求: (假設 **using System;** 敘述已經包含於程式中)
 - (a) 只用一個敘述，將已宣告設值之 **int** 變數 **n** 以遞減算子減 1 後，再設值 (assign) 給他處已宣告之 **int** 變數 **m** (3%)
 - (b) 宣告 **bool** 變數 **r**，並設其值為(非 **p**) "或"(or) **q** 之邏輯演算的結果。
bool 變數 **P** 與 **q** 均已於他處已宣告設值 (3%)
 - (c) 宣告 **double** 變數 **y**，並令其值為他處已宣告設值之 **double** 變數 **t** 與 2π 之積的餘弦函數(cosine)值，注意此處 π 為圓周率 (3%)
 - (d) 宣告 **string** 變數 **state**，利用三元運算子使其在他處已宣告設值之 **double** 變數 **membranePotential** 大於等於 -55 時等於"激發"，反之則等於"靜態" (3%)
 - (e) 宣告變數 **c** 為 **char** 型別，並令其值為倒斜線(back slash)字元 (3%)
3. 撰寫一或數個 C#敘述達成下列要求: (假設 **using System;** 敘述已經包含於程式中)
 - (a) 宣告一個 **int** 常數 **N**，其值為 7 (3%)
 - (b) 宣告一個列數與行數均為 **N** 的 **int** 二維陣列，命名為 **a** (3%)
 - (c) 宣告一個亂數產生器物件 **rand**，種子數為 777 (3%)
 - (d) 將每個 **a** 陣列元素設為一個由 **rand** 產生，在 2(含)與 12(含)之間的亂數 (3%)
 - (e) 計算 **a** 陣列各列的總和，存入一維 **int** 陣列 **rowSum** 的對應元素。令 **rowSum** 已於他處宣告 (3%)
4. 找出以下程式片段之錯誤，並予更正。
 - (a) (3%)
int 1stPlace = 1;

(b) (3%) 以下程式片段應產生三種輸出之一：

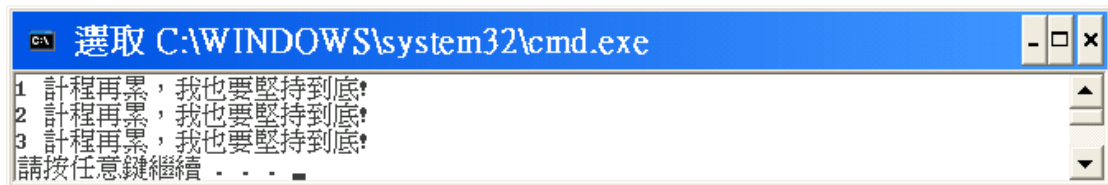
```
char age = Console.ReadLine().ToCharArray()[0];
switch (age)
{
    case '少': Console.WriteLine("紅燭昏羅帳");
    case '壯': Console.WriteLine("斷雁叫西風");
    default: Console.WriteLine("鬢已星星也");
}
```

(c) (3%)

```
int i = 1;
while( i <= 3 )
{
    Console.WriteLine( i/(i-1) );
    ++i;
}
```

(d) (3%) 以下程式片段執行後，應出現如後畫面：

```
for( int i=0; i<3; ++i )
{
    Console.WriteLine(
        "{0} 計程再累，我也要堅持到底!", i++);
}
```



(e) (3%) 下列程式片段應定義兩個函式

```
static void f1(int x)
{
    Console.WriteLine("f1({0})", x);

    static void f2(int x)
    {
        Console.WriteLine("f2({0})", x);
    }
}
```

5. 試寫出下列程式的輸出 (5%)

```
using System;
```

```
namespace MidTerm2011Problem5
```

```
{  
    class Program  
    {  
        static void Main(string[] args)  
        {  
  
            int[] a = { 4, 5, 6 };  
            int[] b = a;  
            WhatDoesThisDo(a);  
            Console.WriteLine("a[2] = {0}, b[2] = {1}",  
                              a[2], b[2]);  
        }  
        static void WhatDoesThisDo(int[] x)  
        {  
            x = new int[4] { 8, 9, 0, 1 };  
            x[2] = 7;  
        }  
    }  
}
```

6. 試寫出下列程式的輸出 (10%)

```
using System;
```

```
namespace MidTerm2011Problem6
```

```
{  
    class Program  
    {  
        static void Main(string[] args)  
        {  
            string pattern = "survey";  
            char[] p = pattern.ToCharArray();  
            string target = "surgery";  
            char[] t = target.ToCharArray();  
            int[,] c = DP(p, t);  
        }  
    }  
}
```

```

    int n = t.Length;
    int m = p.Length;
    int i;
    int j;
    for (i = 0; i <= m; ++i)
    {
        for (j = 0; j <= n; ++j)
        {
            Console.Write(c[i, j] + "\t");
        }
        Console.WriteLine();
    }
}
static int[,] DP(char[] p, char[] t)
{
    int m = p.Length;
    int n = t.Length;
    int[,] c = new int[m+1, n+1];
    int i;
    int j;
    for (j = 0; j <= n; ++j)
    {
        c[0, j] = 0;
    }
    for (i = 0; i <= m; ++i)
    {
        c[i, 0] = i;
    }
    for (i = 1; i <= m; ++i)
    {
        for (j = 1; j <= n; ++j)
        {
            c[i, j] = (p[i-1] == t[j-1]) ?
                c[i - 1, j - 1] :
                1 +
                Minimum(c[i - 1, j], c[i, j - 1],
                    c[i - 1, j - 1]);
        }
    }
}

```

```

    }
    return c;
}
static int Minimum(int x, int y, int z)
{
    if (x <= y && x <= z) return x;
    if (y <= x && y <= z) return y;
    return z;
}
}
}

```

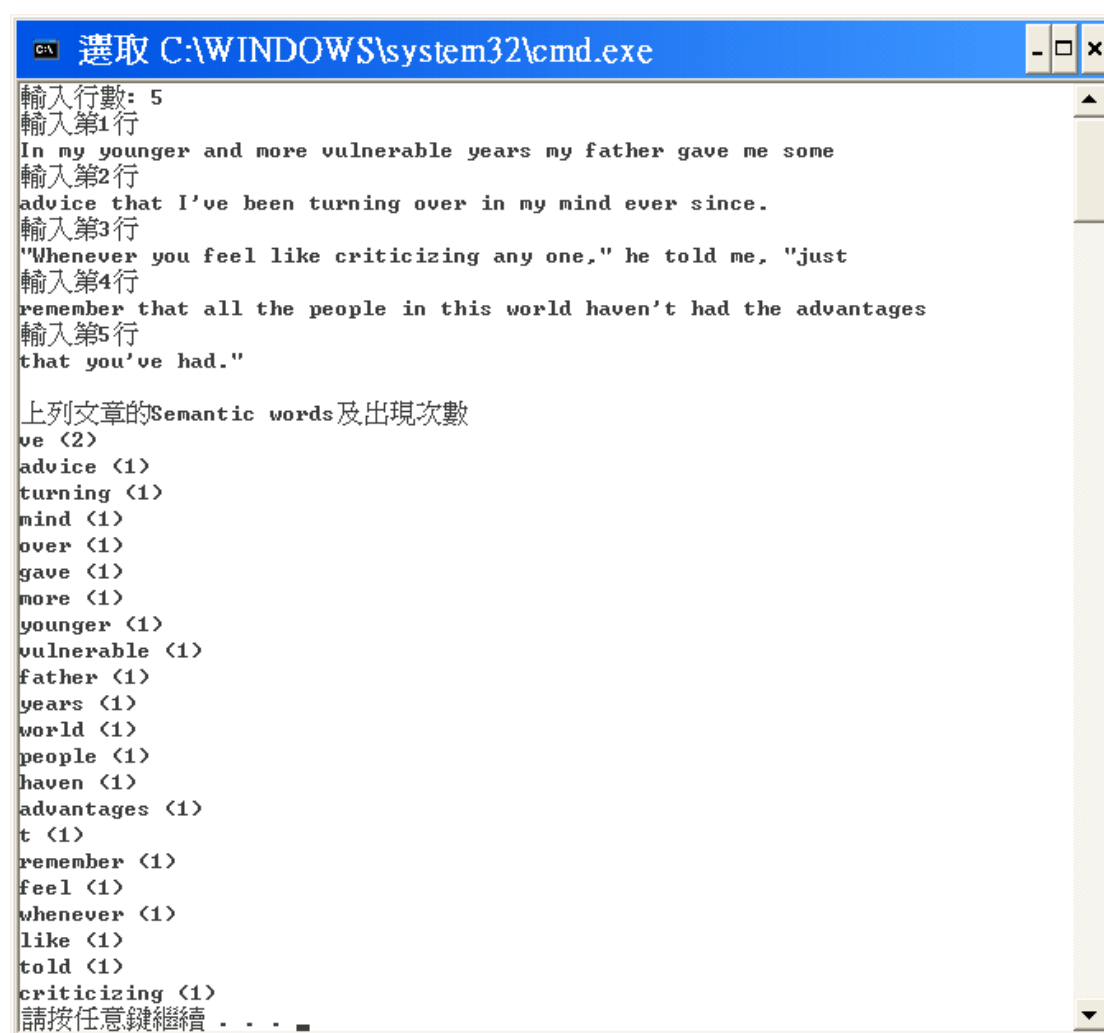
7. 處理英文文件時，常需要把文件中較具意義的單字(semantic words)列出來，並且計算每個這些單字出現的次數。所謂較具意義的單字通常指刪除標點符號及”stop words”之後，留下的單字，其分佈情形可作為文章的特徵(feature)。而”stop words”指文法作用較大，實際意義較少的單字，如冠詞(“a,” “an,” “the”)、連接詞(“and,” “or,” etc.)、代名詞及其所有格(“it,” “his,” etc.)、介詞(“above,” “about,” etc.)、飾詞(“may,” “can,” etc.)、轉折語(“however,” “because,” etc.)、助動詞及經常用語(“is,” “have,” “do,” “make,” “use,” “give,” etc.)、數詞(“one,” “two,” etc.)等。以美國詩人 Robert Frost 著名的詩 The Road Not Taken 之最後一段為例：

I shall be telling this with a sigh
 Somewhere ages and ages hence:
 Two roads diverged in a wood, and I--
 I took the one less traveled by,
 And that has made all the difference

其 stop words 包括 I、shall、be、this、with、a、somewhere、hence、two、in、and、took、the、one、by、that、has、made、all。留下來的 semantic words 為(括弧中的數字為出現次數)：telling (1)、sigh (1)、ages (2)、roads (1)、diverged (1)、wood (1)、less (1)、traveled (1)、difference (1)。網頁 <http://armandbrahaj.blog.al/2009/04/14/list-of-english-stop-words/> 列有一般常用的 stop words。

請寫一個程式，輸入文句，從中找出 semantic words，計算各 semantic word 出現次數，並將結果依出現次數，由高而低排序顯示於主控台螢幕。以美國小說家 F. Scott Fitzgerald 的名著 The Great Gatsby (大亨小傳)第一章的開頭兩

小段為例，對應之主控台螢幕如下圖：



```
選取 C:\WINDOWS\system32\cmd.exe
輸入行數: 5
輸入第1行
In my younger and more vulnerable years my father gave me some
輸入第2行
advice that I've been turning over in my mind ever since.
輸入第3行
"Whenever you feel like criticizing any one," he told me, "just
輸入第4行
remember that all the people in this world haven't had the advantages
輸入第5行
that you've had."

上列文章的Semantic words及出現次數
ve (2)
advice (1)
turning (1)
mind (1)
over (1)
gave (1)
more (1)
younger (1)
vulnerable (1)
father (1)
years (1)
world (1)
people (1)
haven (1)
advantages (1)
t (1)
remember (1)
feel (1)
whenever (1)
like (1)
told (1)
criticizing (1)
請按任意鍵繼續 . . .
```

你可以使用下列已準備好的函式：

輸入文句之函式：

```
static string[] InputArticle()
{
    Console.Write("輸入行數: ");
    int nLines = int.Parse(Console.ReadLine());
    string[] line = new string[nLines];
    int iL;
    for (iL = 0; iL < nLines; ++iL)
    {
        Console.WriteLine("輸入第{0}行", iL+1);
        line[iL] = Console.ReadLine();
    }
}
```

```

    return line;
}

```

由一行文句分離出不含標點符號之單字，可用如下函式：

```

static string[] ProcessLine(string line)
{
    char[] separator =
    { ' ', ',', '.', ':', ';',
      '\'', '\\', '"', '/', '`',
      '-', '|', '(', ')', '[',
      ']', '{', '}', '!', '@',
      '#', '$', '%', '^', '&',
      '*', '+', '_', '=', '~',
      '<', '>', '?'
    };
    string[] word = line.Split(separator);

    // 將單字改為全小寫，以便比對stop words
    int iW;
    for (iW = 0; iW < word.Length; ++iW)
    {
        word[iW] = word[iW].ToLower();
    }
    return word;
}

```

判斷某單字是否 stop words 則可用如下函式：

```

static bool IsAStopWord(string word)
{
    // 不算完全，但可接受的stop words列表
    string[] stopword =
    {
        "", "a", "able", "about", "across",
        "after", "all", "almost", "also", "am",
        "among", "an", "and", "any", "are",
        "as", "at", "be", "because", "been",
        "but", "by", "can", "cannot", "could",
        "dear", "did", "do", "does", "either",
    }
}

```

```

    "else", "ever", "every", "for", "from",
    "get", "gets", "got", "had", "has",
    "have", "he", "hence", "her", "hers",
    "him", "his", "how", "however", "i",
    "if", "in", "into", "is", "it",
    "its", "just", "least", "let", "lets",
    "likely", "made", "make", "makes", "may",
    "me", "might", "most", "must", "my",
    "neither", "no", "nor", "not", "of",
    "off", "often", "on", "one", "only",
    "or", "other", "our", "own", "rather",
    "said", "say", "says", "shall", "she",
    "should", "since", "so", "some", "somewhere",
    "take", "taken", "takes", "than", "that",
    "the", "their", "them", "then", "there",
    "these", "they", "this", "tis", "to",
    "too", "took", "twas", "two", "us",
    "want", "wants", "was", "we", "were",
    "what", "when", "where", "which", "while",
    "who", "whom", "why", "will", "with",
    "would", "yet", "you", "your"
};
bool result = ( Array.IndexOf(stopword, word) >= 0 );
return result;
}

```

上列函式可直接在答案中引用，不需重抄一遍。
 能利用虛擬碼思考，有測試規劃，並有適當註解，善用 **Debug.Assert** 者，
 才可得本題滿分 25 分。(25%)

提示：一開始不知 semantic words 有多少，可宣告足夠大的一維陣列 **sw** 及 **fr**，元素初值分別均設為 **null** 及 **0**。接著逐一找出 semantic words 時，暫時儲存 semantic words 及其出現次數於陣列前端。待 semantic words 全部找出後，假設共有 **nSW** 個，利用

```

Array.Copy(sw, semanticWord, nSW);
Array.Copy(fr, freq, nSW);

```

把 **sw** 及 **fr** 的前 **nSW** 個元素複製到大小為 **nSW** 的一維陣列 **semanticWord** 及 **freq**，以便排序。