

通識計算機程式設計期末考參考解答, 6/19/2009

1. 撰寫 C# 敘述達成下列要求: (假設 `using System;` 及 `using System.IO;` 敘述已經包含於程式中)

(a) 宣告 3 列 2 行之整數二維陣列 `a`(3%)

答：

```
int [,] a = new int[3,2];
```

(b) 將 `a` 的內容設為 $\begin{bmatrix} 2 & 5 \\ -1 & 4 \\ 7 & 3 \end{bmatrix}$ (3%)

答：

```
a[0, 0] = 2;
a[0, 1] = 5;
a[1, 0] = -1;
a[1, 1] = 4;
a[2, 0] = 7;
a[2, 1] = 3;
```

(c) 撰寫迴圈敘述，找出 `a` 內絕對值最大的元素，設定其值為 `aMax`(9%)

答：

```
int i;
int j;
int aMax = 0;
for (i = 0; i < 3; ++i)
{
    for (j = 0; j < 2; ++j)
    {
        if (Math.Abs(a[i, j]) > Math.Abs(aMax))
            aMax = a[i, j];
    }
}
```

(d) 建立一個文字輸出串流(`StreamWriter`)，命名為 `output`，對應檔名為 `Test.dat`(3%)

答：

```
StreamWriter output = new StreamWriter("Test.dat");
```

(e) 在 **output** 中寫入列數 3 與行數 2，及 3 列 2 行之整數二維陣列 **a** 內容，同一列數據之間以一個空白隔開，不同列數據須換行 (9%)

答：

```
output.WriteLine(3 + " " + 2);
for (i = 0; i < 3; ++i)
{
    for (j = 0; j < 2; ++j)
    {
        output.Write(a[i, j] + " ");
    }
    output.WriteLine();
}
```

(f) 關閉 **output**(3%)

答：

```
output.Close();
```

2. 找出以下程式片段之錯誤，並予更正.

(a) (6%) 兩個錯誤

```
class B
{
    protected int i;
    public B(int i)
    {
        this.i = i;
    }
    public int I
    {
        get { return i; }
    }
}
class D : B
{
    private int j;
    public D(int i, int j) : B(i) ← 改為 base(i)才合語法
    {
        this.j = j;
    }
}
```

```

    }
    public int Sum
    {
        get { return i+j; }
    }
}
class Program
{
    static void Main(string[] args)
    {
        D d = new D(3, 5);
        Console.WriteLine(d.i); ← i 是protected成員變數
                                不能在類別或子類別外使
                                用，應改為 d.I
    }
}

```

(b) (6%) 兩個錯誤

class A ← 改為 abstract class A

```

{
    public A() { }
    abstract public int Test();
}
class Program
{
    static void Main(string[] args)
    {
        A a = new A(); ←不能建立abstract物件
    }
}

```

也可刪除 **abstract** (只有一個錯誤)

(c) (3%).一個錯誤

```

try
{
    Console.Write("輸入除數: ");
    int number = int.Parse(Console.ReadLine());
}

```

```

        int result = 5 / number;
    }
    catch (Exception e)
    {
        Console.WriteLine(e);
    }
    catch (DivideByZeroException e)
    {
        Console.WriteLine(e);
    }

```

← 較特別的 **Exception** 物件要在較一般的 **Exception** 物件前被 **catch** 應改為

```

    catch (DivideByZeroException e)
    {
        Console.WriteLine(e);
    }

    catch (Exception e)
    {
        Console.WriteLine(e);
    }

```

(d) (3%) 一個錯誤

```

using System;
using System.IO;
using System.Runtime.Serialization;
using System.Runtime.Serialization.Formatters.Binary;
. . . . .

```

class B → 欲序列化之物件類別，需註明 **Serializable**

應改為

[Serializable]

class B

```

{
    protected int i;
    public B(int i)

```

```

    {
        this.i = i;
    }
    public int I
    {
        get { return i; }
    }
}
class Program
{
    static void Main(string[] args)
    {
        B b = new B(3);
        BinaryFormatter formatter = new
            BinaryFormatter();
        try {
            FileStream output = new FileStream("Test.dat",
                FileMode.Create, FileAccess.Write );
            formatter.Serialize(output, b);
            output.Close();
        }
        catch (SerializationException)
        {
            Console.WriteLine(
                "Error writting to file in output");
        }
        catch (IOException)
        {
            Console.WriteLine(
                "Can not create or close file for output");
        }
    }
}

```

(e) (6%) 一組錯誤

```

using System;
using System.Collections.Generic;
. . . . .

```

class B → 要使用List的Sort功能，必須實作IComparable介面
應改為 **class B : IComparable**

```
{
    protected int i;
    public B(int i)
    {
        this.i = i;
    }
    public int I
    {
        get { return i; }
    }
    → 應增加
    public int CompareTo(B b)
    {
        int result = 0;
        if (this.i < b.i)
        {
            result = -1;
        }
        else if (this.i > b.i)
        {
            result = 1;
        }
        else
        {
            result = 0;
        }
        return result;
    }
}
class Program
{
    static void Main(string[] args)
    {
        List<B> lb = new List<B>();
        lb.Add(new B(5));
        lb.Add(new B(2));
    }
}
```

```

        lb.Add(new B(3));
        lb.Sort();
    }
}

```

3. 試寫出下列程式的輸出 (15%)

```

using System;
namespace Final2009Problem3
{
    class List
    {
        private int n;
        private int[] x;
        public List(int[] y)
        {
            n = y.Length;
            x = new int[n];
            int i;
            for (i = 0; i < n; ++i)
            {
                x[i] = y[i];
            }
        }
        public void QuickSort()
        {
            Quick(0, n-1);
        }
        private void Quick(int lb, int ub)
        {
            int j;
            if (lb < ub)
            {
                Rearrange(lb, ub, out j);
                Quick(lb, j - 1);
                Quick(j + 1, ub);
            }
        }
    }
}

```

```

private void Rearrange(int lb, int ub, out int j)
{
    int a = x[lb];
    j = lb;
    int up = ub;
    int down = lb;
    do
    {
        while (up > down && x[up] >= a) up--;
        j = up;
        if (up != down)
        {
            x[down] = x[up];
            while( down < up && x[down] <= a) down++;
            j = down;
            if (down != up) x[up] = x[down];
        }
    } while (up != down);
    x[j] = a;
    Dump();
}

public void Dump()
{
    int i;
    for (i = 0; i < n; ++i )
    {
        Console.Write(x[i] + "\t");
    }
    Console.WriteLine();
}

}

class Program
{
    static void Main(string[] args)
    {
        int[] x = { 25, 57, 48, 37, 12, 92, 86, 33 };
        List ls = new List(x);
        ls.QuickSort();
    }
}

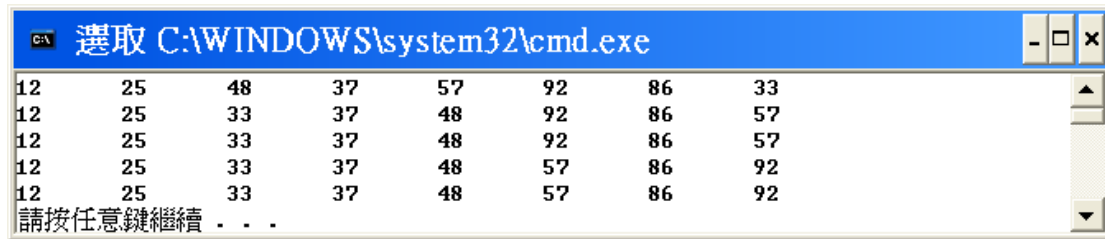
```

```

    }
}
}

```

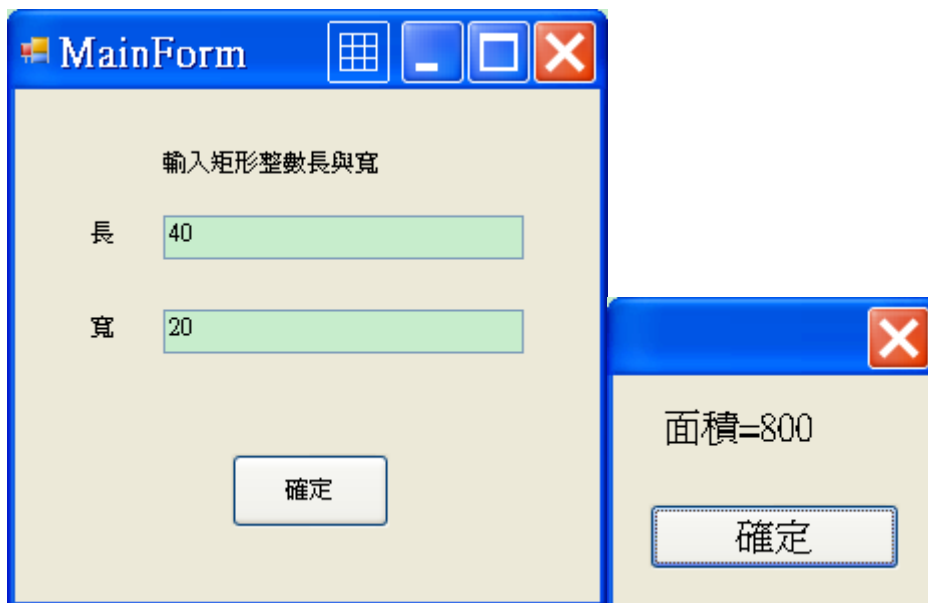
答：



4. 依據以下描述及程式框架，完成指定程式。你在答案卷只需寫下程式註解標示為(a)到(e)的部份。各部份配分分別註明，總共 6%。

程式描述:

建立如下視窗介面，由使用者輸入矩形之整數長與寬，建立對應矩形物件，再以訊息盒顯示矩形面積。



```

// 檔案 MainForm.cs
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;

```

```

using System.Linq;
using System.Text;
using System.Windows.Forms;

namespace Final2009Problem4
{
    public partial class MainForm : Form
    {
        public MainForm()
        {
            InitializeComponent();
        }

        private void button_click(object sender, EventArgs e)
        {
            /*******
            /* (a) 加入敘述，由textbox1及textbox2取得矩形
               長與寬(1%)*
            /* (b) 加入敘述，建立對應矩形物件 (1%)*
            /* (c) 加入敘述，取得對應矩形物件面積 (1%)*
            /* (d) 加入敘述，以訊息盒顯示對應矩形物件之面積(1%)*
            /*******
        }
    }
}

```

```

//檔案 Rectangle.cs
using System;

```

```

namespace Final2009Problem4
{
    public class Rectangle
    {
        /* (e) 完成矩形類別之宣告(2%) */
    }
}

```

答：

(a)

```
int width = Convert.ToInt32(textBox1.Text);  
int height = Convert.ToInt32(textBox2.Text);
```

(b)

```
Rectangle rec = new Rectangle(width, height);
```

(c)

```
int area = rec.Area();
```

(d)

```
MessageBox.Show("面積="+area.ToString());
```

(e)

```
private int width;  
private int height;  
private int area;  
public Rectangle(int width, int height)  
{  
    this.width = width;  
    this.height = height;  
    area = width * height;  
}  
public int Area()  
{  
    return area;  
}
```

5. 鄭老師清點所收藏的圖書與電影 DVD，得到圖書資料如表 1，電影 DVD 資料如表 2。

表 1. 藏書資料

seqNo	title	majorAuthor	publisher	year
001	哈利波特:消失的密室	J. K. 羅琳	皇冠	2001
003	墨水心	柯奈莉亞·馮克	大田	2005

表 2. 電影 DVD 資料

seqNo	title	majorDirector	MajorActor	distributor	year
-------	-------	---------------	------------	-------------	------

002	變形金剛	麥可貝	席亞拉伯夫	得利影視	2009
-----	------	-----	-------	------	------

撰寫程式依 seqNo 順序顯示各項收藏資料，以上述之假想情境為測試場景。不使用類別者，最高得 13 分；使用類別，不使用多型者，最高得 20 分；使用類別及多型，但未使用 **Collections** 者，最高得 24 分；正確使用使用類別、多型、**Collections** 者，最高得 25 分。使用 C#資料庫功能者，依使用正確程度給分。

(25%)

答：

```
using System;
using System.Collections.Generic;

abstract public class Item
{
    protected string seqNo;
    protected string title;
    protected int year;
    public Item(string seqNo, string title, int year)
    {
        this.seqNo = seqNo;
        this.title = title;
        this.year = year;
    }
    abstract public void Dump();
}

public class Book : Item
{
    private string majorAuthor;
    private string publisher;
    public Book(string seqNo, string title,
        string majorAuthor, string publisher, int year) :
        base( seqNo, title, year )
    {
        this.majorAuthor = majorAuthor;
        this.publisher = publisher;
    }
}
```

```

public override void Dump()
{
    Console.Write(seqNo + "\t");
    Console.Write(title + "\t");
    Console.Write(majorAuthor + "\t");
    Console.Write(publisher + "\t");
    Console.WriteLine(year);
}
}

public class Video : Item
{
    private string majorDirector;
    private string majorActor;
    private string distributor;
    public Video(string seqNo, string title,
        string majorDirector, string majorActor,
        string distributor, int year)
        : base(seqNo, title, year)
    {
        this.majorDirector = majorDirector;
        this.majorActor = majorActor;
        this.distributor = distributor;
    }
    public override void Dump()
    {
        Console.Write(seqNo + "\t");
        Console.Write(title + "\t");
        Console.Write(majorDirector + "\t");
        Console.Write(majorActor + "\t");
        Console.Write(distributor + "\t");
        Console.WriteLine(year);
    }
}

namespace Final2009Problem5
{
    class Program

```

```

{
    static void Main(string[] args)
    {
        List<Item> list = new List<Item>();
        list.Add( new Book( "001", "哈利波特:消失的密室",
            "J. K. 羅琳", "皇冠", 2001 ) );
        list.Add(new Video("002", "變形金剛", "麥可貝",
            "席亞拉伯夫", "得利影視", 2009));
        list.Add( new Book( "003", "墨水心",
            "柯奈莉亞.馮克", "大田", 2005 ) );
        foreach( Item item in list )
        {
            item.Dump();
        }
    }
}

```