



物件導向程式設計實習期末報告

資訊一乙 陳維基



目錄：

一 學習內容：

1. C#基本語法 與 Visual Studio
2. Visual Studio 元件
3. Class、物件建立
4. 連接資料庫
5. 資料庫的增刪查改

二 學習成果：

1. 期末專案 – 俄羅斯方塊

三 心得



學習內容 – C#基本語法 與 Visual Studio

C#的語法與上學期所學的C++差不多，所以在程式撰寫上並沒有遇到太大的問題。不過在撰寫時有發現一些很方便的內建的函式，是原先C++內沒有用過的，例如：`List.Contains()`、`int.Parse()`...等。

最需要了解的是Visual Studio的環境，這學期主要是學習Windows Forms App的專案類型，要理解各個視窗的作用，了解除錯的方法。

這學期學這個跟上學期最大的差別 – 介面，這學期所學的是有介面的，畫面上的編排，也是這次學到比較不一樣的東西。



學習內容 — Visual Studio 元件

而比較新的東西大概就是Visual Studio內的元件了吧！要認識每種元件的性質即使用方法，在了解不同的元件上就花了不少時間。

各元件呼叫的方法已有很大的學問，包括：事件、屬性...等等，新增不同的事件，事件的種類也是不枚勝舉，因此了解事件的用法也是費盡了心思。

之後又花了時間研究自動生成元件的方法，包括將生成出來的元件新增事件、定位...等。



學習內容 – Class、物件建立

Class的部分學C++時我並不常用到，所以對這部分的掌握度並不高，不過在這學期Class的使用機會大大的提升，讓我學習到了Class的建構子、函式...等等。

在紀錄資料時Class就很方便，還可以儲存在Array或List裡，再搭配前面所講的一些函式就可達到許多效果，例如：判斷資料是否重複、排序資料...等。



學習內容 — 連接資料庫

SQL資料庫，是這學期主要學習的資料庫，包括：建立資料庫、資料表...等。這學期所學的連接資料庫的方法有兩種：(1) SQL語法 (2) 建立Entity使用Lambda語法連接，這兩種。

其中在寫SQL語法時寫起來都會很卡，主要是因為他都是字串，所以在撰寫時都要慢慢打，而且又很難記，常常都要重新看怎麼寫。

而建立Entity的方法，有個優點 – 他有提示字，意味著不需要記整個字串怎麼打，在撰寫程式時也比較快速，雖然在執行時SQL語法相對比較快，不過在使用體驗上並沒有遇過執行很久的情況。



學習內容 — 資料庫的增刪查改

我主要是使用Entity來做增刪查改，在建Entity時有發現，每一張資料表的架構都是Class，所以每一張表都可以當作Class來使用，可以建立先的物件。

新增：建立新的物件，在把它放進資料庫。

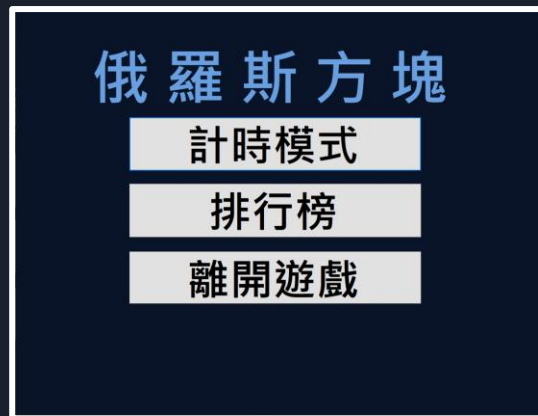
刪除：`db.Remove()`

查詢：`db.Select()`

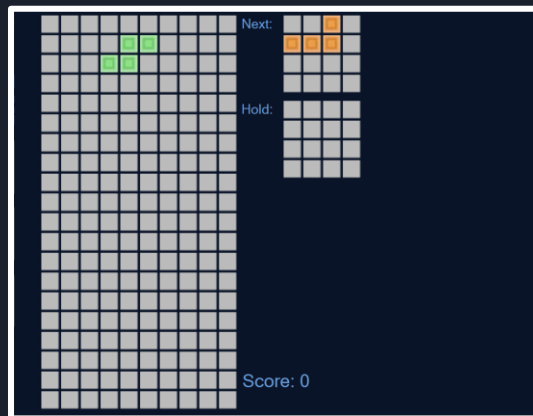
更改：透過查詢出來的值，來更改底下的屬性。

Lambda語法有很多很方便的方法來查詢、分類，所以可以做到很多功能。

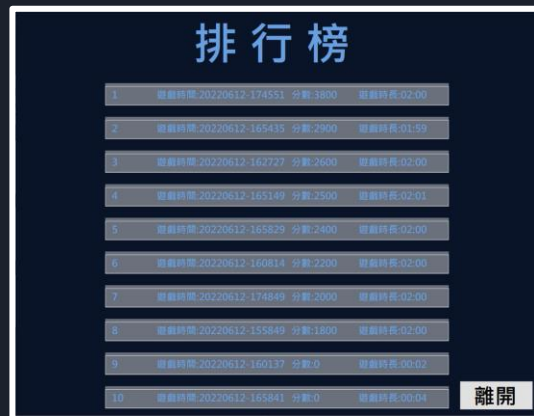
學習成果 — 期末專案 — 俄羅斯方塊



主畫面



遊戲畫面



排行榜畫面

學習成果 — 期末專案 — 俄羅斯方塊

主畫面所用到的技術：

1. 使用panel切換畫面
2. 將主畫面傳至其他畫面，做出可切換頁面的功能
3. SoundPlayer在按下按鈕時播放指定音效

6 個參考

```
public void showform(Form form)
{
    panel1.Controls.Clear();
    form.FormBorderStyle = FormBorderStyle.None;
    form.TopLevel = false;
    panel1.Controls.Add(form);
    form.Location = new Point(0, 0);
    Focus();
    form.Show();
}
```

切換畫面的函式



學習成果 — 期末專案 — 俄羅斯方塊

遊戲畫面所用到的技術：

1. 建立Class記錄不同類型的俄羅斯方塊
2. 自動生成元件，並使用List紀錄之
3. 偵測鍵盤按鍵事件
4. Timer做出方塊墜落的效果，改變定位點來清除方塊，及重新生成方塊
5. SoundPlayer在對方塊操作時播放指定音效
6. PictureBox的應用，來變更方塊
7. Random生成下一個方塊
8. StreamWriter將成績輸出成文字檔
9. WindowsMediaPlayer播放遊戲BGM

學習成果 — 期末專案 — 俄羅斯方塊

```
18 個參考
class block
{
    public Point[] blockPoint = new Point[4] { new Point(), new Point(), new Point(), new Point() };
    public string blocktype;
    public int blockstyle;
}
```

俄羅斯方塊的Class

blockPoint為方塊的相對座標

blocktype為方塊類型

blockstyle為方塊方向

```
case "I":
    switch (style)
    {
        case 1:
            blockPoint[0] = new Point(-1, 0);
            blockPoint[1] = new Point(0, 0);
            blockPoint[2] = new Point(1, 0);
            blockPoint[3] = new Point(2, 0);
            break;
        case 2:
            blockPoint[0] = new Point(1, -1);
            blockPoint[1] = new Point(1, 0);
            blockPoint[2] = new Point(1, 1);
            blockPoint[3] = new Point(1, 2);
            break;
        case 3:
            blockPoint[0] = new Point(-1, 1);
            blockPoint[1] = new Point(0, 1);
            blockPoint[2] = new Point(1, 1);
            blockPoint[3] = new Point(2, 1);
            break;
        case 4:
            blockPoint[0] = new Point(0, -1);
            blockPoint[1] = new Point(0, 0);
            blockPoint[2] = new Point(0, 1);
            blockPoint[3] = new Point(0, 2);
            break;
    }
    break;
```

利用Class函式設定值

學習成果 — 期末專案 — 俄羅斯方塊

排行榜畫面所用到的技術：

1. StreamReader讀取文字檔
2. 自動生成元件

```
GroupBox g = new GroupBox();
g.SetBounds(200, 160 + count*75, 766, 50);
g.BackColor = Color.FromArgb(100, 255, 255, 255);
Controls.Add(g);

Label no = new Label();
no.SetBounds(10, 12, 100, 30);
no.Text = (++count).ToString();
no.Font = new Font("微軟正黑體", 16, FontStyle.Bold);
no.BackColor = Color.FromArgb(0, 0, 0, 0);
no.ForeColor = Color.FromArgb(105, 160, 225);
g.Controls.Add(no);

Label time = new Label();
time.SetBounds(110, 12, 300, 30);
time.Text = "遊戲時間:" + (data.time).ToString("yyyMMdd-HHmss");
time.Font = new Font("微軟正黑體", 16, FontStyle.Bold);
time.BackColor = Color.FromArgb(0, 0, 0, 0);
time.ForeColor = Color.FromArgb(105, 160, 225);
g.Controls.Add(time);

Label score = new Label();
score.SetBounds(410, 12, 150, 30);
score.Text = "分數:" + (data.score).ToString();
score.Font = new Font("微軟正黑體", 16, FontStyle.Bold);
score.BackColor = Color.FromArgb(0, 0, 0, 0);
score.ForeColor = Color.FromArgb(105, 160, 225);
g.Controls.Add(score);

Label gameTime = new Label();
gameTime.SetBounds(560, 12, 196, 30);
gameTime.Text = "遊戲時長:" + (data.gameTime).ToString();
gameTime.Font = new Font("微軟正黑體", 16, FontStyle.Bold);
gameTime.BackColor = Color.FromArgb(0, 0, 0, 0);
gameTime.ForeColor = Color.FromArgb(105, 160, 225);
g.Controls.Add(gameTime);
```

自動生成元件



學習成果 — 期末專案 — 俄羅斯方塊

這次在寫這個專案的時候，花最多時間的是想如何去移動方塊，最後的解法是：移動一個基準點，再從它周圍生成方塊。

檢查每一行是用PictureBox的屬性Tag去做的，判斷List內，所有的PictureBox的Tag值來清除方塊並移動其他方塊。



心得

這學期真的學了很多之前沒接觸的東西，原本C++是沒有畫面的，這學期透過Visual Studio來做出畫面的呈現，讓專案跟上學期相比有了更多的發揮空間，雖然在適應編譯環境花了些時間，也接觸了許多沒碰過或不熟悉的領域，包括：Class、物件、元件、建文字檔、資料庫...等。之後也會想嘗試做出不同的專案來精進自己的程式撰寫能力及熟悉度。