## 數字對對碰

財金一 B09703082 何宗穎

# 一、動機

翻牌對對碰是我小時候使用智障型手機時最喜歡玩的遊戲。日前,靠著課 堂講義上完成井字遊戲之程式實作之後,我便回想起關於翻牌對對碰的美好回 憶。經過初步評估,我認為它的程式撰寫難度和井字遊戲差不多,因此打算藉 著這次作業撰寫簡易版的翻牌對對碰遊戲,來重溫童年的美好時光。

#### 二、構想解說

我將此遊戲取名為「數字對對碰」;此遊戲共包含 16 張卡片、1~8 共 8 個數字,每個數字皆被隨機分配至 2 張卡片背後。遊戲流程為,玩家藉由輸入卡片座標來翻開卡片,一窺卡片背面的數字,而當玩家輸入下一組座標之後,先前翻開之卡片便會轉回正面,因此,玩家有熟記卡片背面的數字之必要。當玩家連續兩次翻開數字相同之卡片,此二張卡片便會永久地轉向正面,玩家成功碰了 1 組數字。此遊戲之獲勝條件為,玩家成功碰出 8 組數字。

至於此遊戲程式架構,大致如下頁圖1。

設定一全為「?」之二維陣列 (陣列 A)



將 1~8 (每個數字各兩次) 隨機分配至 16 張卡片,並將其設定為二維陣列 (陣列 B)



讀取使用者輸入之座標並清空 cmd 之畫面



判斷使用者輸入之座標是否於範圍 (0~3)內,不在則請其重新輸入



判斷此次翻取是否成功碰出數字,是的話,則改變陣列 A。 將陣列 A 內此數字於陣列 B 內的兩個相應位置,其「?」改作陣列 B 相應之數字。



印出目前對對碰之情況。除了使用者此次翻取之卡片為陣列 B 之數字,其餘卡片皆顯示陣列 A 的「?」或數字。



紀錄使用者此次輸入之座標,以便與下一次輸入整合判斷是否成功碰出數字。



判斷已碰出之數字是否已達8組

遊戲結束,使用者獲勝

# 三、程式列表

...(省略系統產生之程式片段)

```
//設定兩棋盤(二維陣列), boardGuess全為卡片背面;
char[,] board = {{ '?', '?', '?', '?' }, { '?', '?', '?' }, { '?', '?', '?', '?' }, { '?', '?', '?', '?' },
Random rand = new Random();
int k;
int io;
int jo;
for (int ans = 1; ans < 9; ans++)
    int loopCounter = 0;
    while (loopCounter < 2)
         k = rand.Next() % 16;
         io = k / 4;
         jo = k \% 4;
         if (board[io, jo] == '?')
             board[io, jo] = char.Parse(ans.ToString());
             loopCounter++;
         }
    }
}
//印出遊戲起始畫面
Console.WriteLine("歡迎來到數字對對碰!");
Console.WriteLine();
Console.WriteLine("
                       0 1 2
                                          ");
Console.WriteLine("---|---|---|---|-");
for (int i = 0; i < 4; i++)
{
    Console.Write(" " + i);
    for (int j = 0; j < 4; j++)
    {
         Console.Write(" | " + "?");
    Console.WriteLine(" | ");
    Console.WriteLine("---|---|---|-");
}
int discovered = 0; //已碰出之數字組數
int counter = 0; //紀錄使用者總共猜了幾次
int ilnput; //使用者此次輸入之座標
int ilnput;
int prelInput = 4; //使用者前次輸入之座標
int preJInput = 4;
while (discovered < 8)
    //讀取使用者輸入,並清空cmd之畫面
    Console.WriteLine();
    Console.Write("請輸入欲翻取卡片之座標,以逗點分隔:");
    string[] input = Console.ReadLine().Split(',');
    iInput = int.Parse(input[0]);
    jInput = int.Parse(input[1]);
    Console.Clear();
```

```
//判斷使用者輸入之座標是否於範圍(0~3)內,不在則請其重新輸入
    if (iInput <0 | iInput >3 | jInput<0 | jInput > 3)
        Console.WriteLine("座標不在範圍內!");
        Console.WriteLine();
                              0 1 2 3
        Console.WriteLine('
        Console.WriteLine("---|---|---|-");
        for (int i = 0; i < 4; i++)
            Console.Write(" " + i);
            for (int j = 0; j < 4; j++)
                 Console.Write(" | " + boardGuess[i, j]);
            Console.WriteLine(" | ");
            Console.WriteLine("---|---|---|---|;
        continue;
    }
    //判斷此次翻取是否成功碰出數字,是的話,則改變boardGuess。
    //將boardGuess內此數字於board內的兩個相應位置,其「?」改作相應之數字。
    if (prelInput < 4)
    {
        if ((board[prelInput, preJInput] == board[iInput, jInput])
           &((prelInput != iInput) | (preJInput != jInput))
           & (boardGuess[iInput, jInput] == '?'))
       {
            boardGuess[prelInput, preJInput] = board[prelInput, preJInput];
            boardGuess[iInput, jInput] = board[prelInput, preJInput];
            discovered++;
        }
    }
    //印出目前目前對對碰之情況與使用者此次翻取之卡片。
   //除了使用者此次翻取之卡片印出board之數字,其餘卡片皆顯示boardGuess之情況。Console.WriteLine(" 0 1 2 3 ");
    Console.WriteLine("---|---|---|---|;
    for (int i = 0; i < 4; i++)
    {
        Console.Write(" " + i);
        for (int j = 0; j < 4; j++)
            if (i == iInput & j == jInput)
            {
                 Console.Write(" | " + board[i, j]);
            }
            else
            {
                 Console.Write(" | " + boardGuess[i, j]);
            }
        Console.WriteLine(" | ");
        Console.WriteLine("---|---|---|---|---|;
   }
    //紀錄將使用者此次翻取之卡片座標,以便與下一次輸入整合判斷是否成功碰出數字。
    prelinput = iinput;
    preJInput = jInput;
    counter++;
    if (discovered!=8)
    {
        Console.WriteLine();
        Console.WriteLine("您已翻取{0}次、碰中{1}組數字", counter, discovered);
   }
Console.WriteLine();
Console.WriteLine("您共花費{0}次碰中8組數字,恭喜獲勝!", counter);
```

## 四、程式測試規劃

本次程式測試規劃如下:

- 1、輸入範圍外之座標。
- 2、連續兩次輸入相同座標。
- 3、連續輸入已碰出之數字的兩個座標。
- 4、完成數字對對碰。

## 五、程式測試執行成果

1、輸入範圍外之座標:

測試結果如圖 2 及圖 3,程式告訴使用者座標不在範圍內,請其重新輸入,並且錯誤座標之輸入並不會增加翻取次數,測試成功。

C:\WINDOWS\system32\cmd.exe	C:\WINDOWS\system32\cmd.exe	
<b>欢迎來到數字對對碰!</b>	座標不在範圍內!	
0 1 2 3	0 1 2 3	
-		
0   2   2   2   2	0   2   2   2   2	
1   2   2   2   2   2	1   2   2   2   2	
2   2   2   2   2   2	2   2   2   2   2	
	-	
3   2   2   2   2   2	3   2   2   2   2	
蟾蜍人欲翻取卡片之座標,以短點分隔:0.4	請輸入欲翻取卡片之座標,以短點分隔	



圖 2

圖 3

#### 2、連續兩次輸入相同座標:

測試結果如圖 4 及圖 5,不論重複輸入之座標為已碰出或是未碰出之數字,雖其數字相同,但程式都不會將其視為成功碰出數字,測試成功。





圖 4

#### 3、連續輸入已碰出之數字的兩個座標:

測試結果如圖 6,雖其數字相同,但程式都不會將其視為成功碰出數字,測試成功。



圖 6

#### 4、完成數字對對碰:

過程中並無差錯,測試成功,僅擷取最後幾步如圖7。

C:\WINDOWS\system3Z\cmd.exe	C\WINDOWS\system32\cmd.exe	on CAWINDOWS\system3Z\cmd.exe
0 1 2 3 0 7 8 7 18 1 1 1 4 1 2 1 7 1 2 1 1 1 1 2 1 1 1 1 1 1 1 1 1	0 1 2 3 0   5   8   7   8   1 1   4   2   7   2   2   2   2   2   3   1   3   7   3   3   3   3   3   3   3   3	0 1 2 3 0   5   8   7   8   1   4   2   7   2   2   4   1   5   7   3   1   3   7   3   1   1   2   7   2
您已翻取28次、碰中5組數字	您已翻取29次、碰中6組數字	您已翻取30次、碰中6组數字
請輸入欲翻取卡片之座標,以提點分隔:0,0。	請輸入欲翻取卡片之座標,以短點分隔:	請輸入欲翻取卡片之座標,以逗點分隔:0,2_
C:\WINDOWS\system3Z\cmd.exe	C\WINDOWS\system32\cmd.exe	C:\WINDOWS\system32\cmd.exe
0 1 2 3 0   5   8   7   8   1   4   2   7   2   2   4   1   5   7   3   1   3   7   3	0 1 2 3 0 1 5 1 8 1 7 1 8 1 1 4 1 2 1 6 1 2 2 1 4 1 1 1 5 7 3 1 1 1 3 1 7 1 3	0 1 2 3 0   5   8   7   8   
您已翻取31次、碰中7組數字 濟輸入欲翻取卡片之座標、以短點分隔:1,2	您已翻取32次、碰中7组數字 請輸入欲翻取卡片之座標,以逗點分隔:3.2。	您共花費33次碰中8組數字,恭喜獲勝! 請按任意鍵繼續

圖 7