

Week8 (4/16)

期中考複習四、第7題

參考作法

r09941007

前言

”參考作法”就是只要參考就好

當然每個人可以有一套自己的程式邏輯&流程
但在進行Project時，**程式必須讓其他人也讀懂**，
故”常見”的規劃方式、規則，自然也較易讓他人理解。

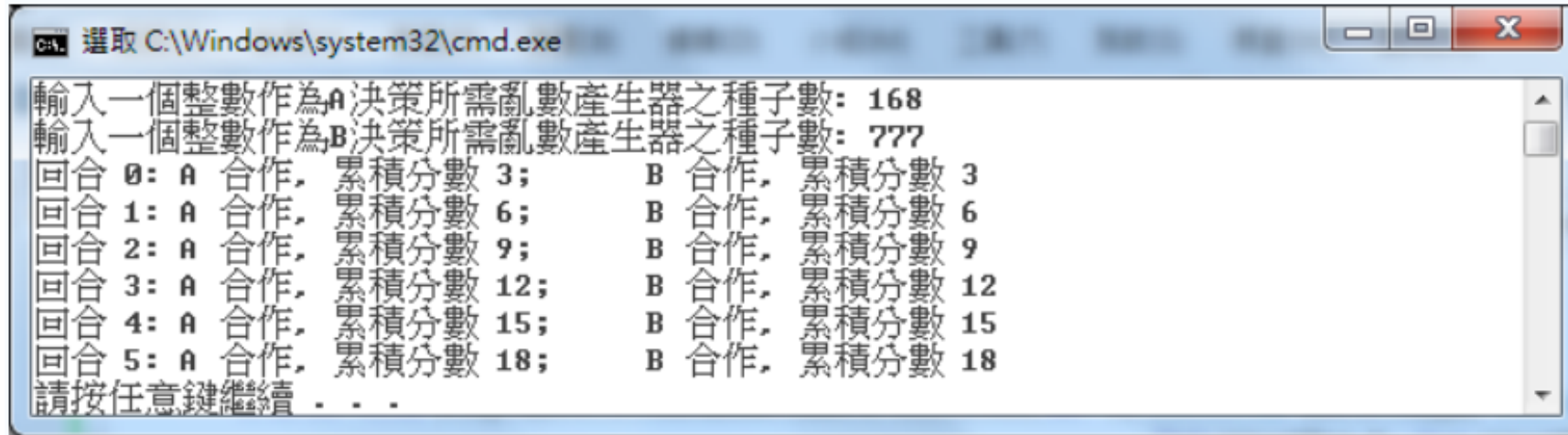
學期初的變數命名規則亦是同樣的道理。。

本題請你撰寫一個 C# 程式，模擬「重複囚徒困境」的狀況，起始雙方均合作(算是第 0 回合)，接著對抗 N 個回合。由第 0 回合開始，在螢幕顯示每一回合雙方採用的策略及由開始到這一回合的累計得分。假設 N 是一個 1(含)與 1000(含)之間的隨機亂數(因此雙方事先不知道要進行多少回合，亂數產生器可以不設種子數(seed)，讓計算機自行決定，使程式每次執行結果都不相同)，且對抗雙方(代號為 A, B)都採用「寬恕地以牙還牙」策略，寬恕對手背叛的機率 A 為 5%，B 為 2%，而其每一回合的酬報矩陣(payoff matrix, 顯示結果的表格)如下：

	B 合作	B 背叛
A 合作	A, B 均得 3 分	A 得 0 分, B 得 5 分
A 背叛	A 得 5 分, B 得 0 分	A, B 均得 1 分

提示：對手背叛時，可以產生一個 0 到 99 之間的隨機亂數，如果這個亂數是 0、1，就寬恕對方(選取「合作」)，否則就選「背叛」(以牙還牙)，這樣即可以用 2% 的機率寬恕對方。類似的作法可以得到 5% 的寬恕機率。兩個對手的亂數產生器種子數可隨意用兩個你自己選定的不同整數。

程式螢幕互動範例:



```
C:\Windows\system32\cmd.exe
輸入一個整數作為A決策所需亂數產生器之種子數: 168
輸入一個整數作為B決策所需亂數產生器之種子數: 777
回合 0: A 合作, 累積分數 3;      B 合作, 累積分數 3
回合 1: A 合作, 累積分數 6;      B 合作, 累積分數 6
回合 2: A 合作, 累積分數 9;      B 合作, 累積分數 9
回合 3: A 合作, 累積分數 12;     B 合作, 累積分數 12
回合 4: A 合作, 累積分數 15;     B 合作, 累積分數 15
回合 5: A 合作, 累積分數 18;     B 合作, 累積分數 18
請按任意鍵繼續 . . .
```

本題滿分 25 分，全部程式集中寫成一個大 **Main** 函式，不區分函式者，最高得 20 分；善用函式，乃至尚未教到的物件導向程式設計(object-oriented programming)者，最高得 23 分；能利用虛擬碼或流程圖思考，適當劃分函式或類別(class)者，最高得 25 分(使用虛擬碼)或 24 分(使用流程圖)。(25%)

第一步

思考程式需具備的功能、使用的物件/變數

變數：

回合數、分數、兩人的選擇、兩人的寬恕機率

物件(class)：

Prisoner：用於代表兩位玩家AB。

功能(method)：

Game - 判斷兩人的決定，所產生的結果。AB進行加減分，並印出結果

SetFRate - 讓我們設定兩人的寬恕機率

SetSeed - 題目要求，藉由給定Seed比對結果是否有Logical error

Action - 讓AB根據對方的選擇做出選擇

第二步-1

規劃各個變數、功能、會在哪邊被用到

(代表哪邊會用到)

變數：

回合數(Game, Main)、分數(Game, Prisoner)、兩人的選擇(Game, Prisoner)、兩人的寬恕機率(Action)

功能(method)：

Game(Main)

SetFRate(Prisoner)

SetSeed(Prisoner)

Action(Prisoner, Game)

規劃程式的Layer

回合數(Game, Main)、分數(Game, Prisoner)、兩人的選擇(Game, Prisoner)、兩人的寬恕機率(Action)

可以看到，上面四個變數，只有回合數跟Prisoner-AB無關，其他都是綁定AB。

=> 分數、選擇、寬恕機率 都寫在class Prisoner內就好

Game- 需要讀入AB兩個Prisoner，會用到他們的選擇、Action、改AB的分數

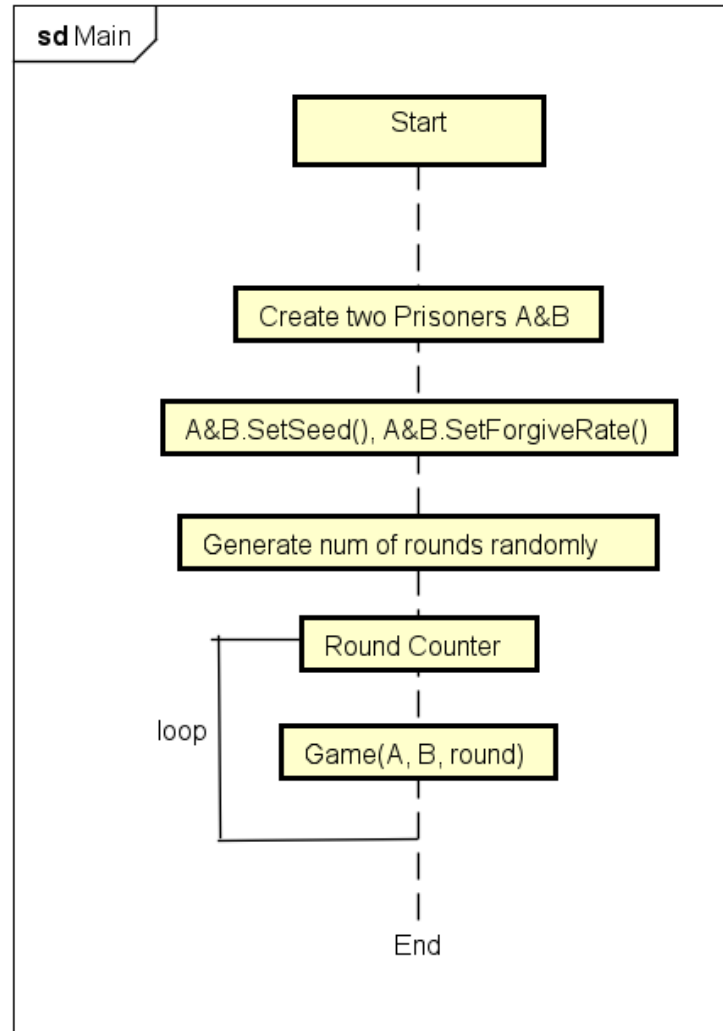
SetFRate(Prisoner) - 只跟Prisoner有關，寫在Prisoner裡面

SetSeed(Prisoner) - 只跟Prisoner有關，寫在Prisoner裡面

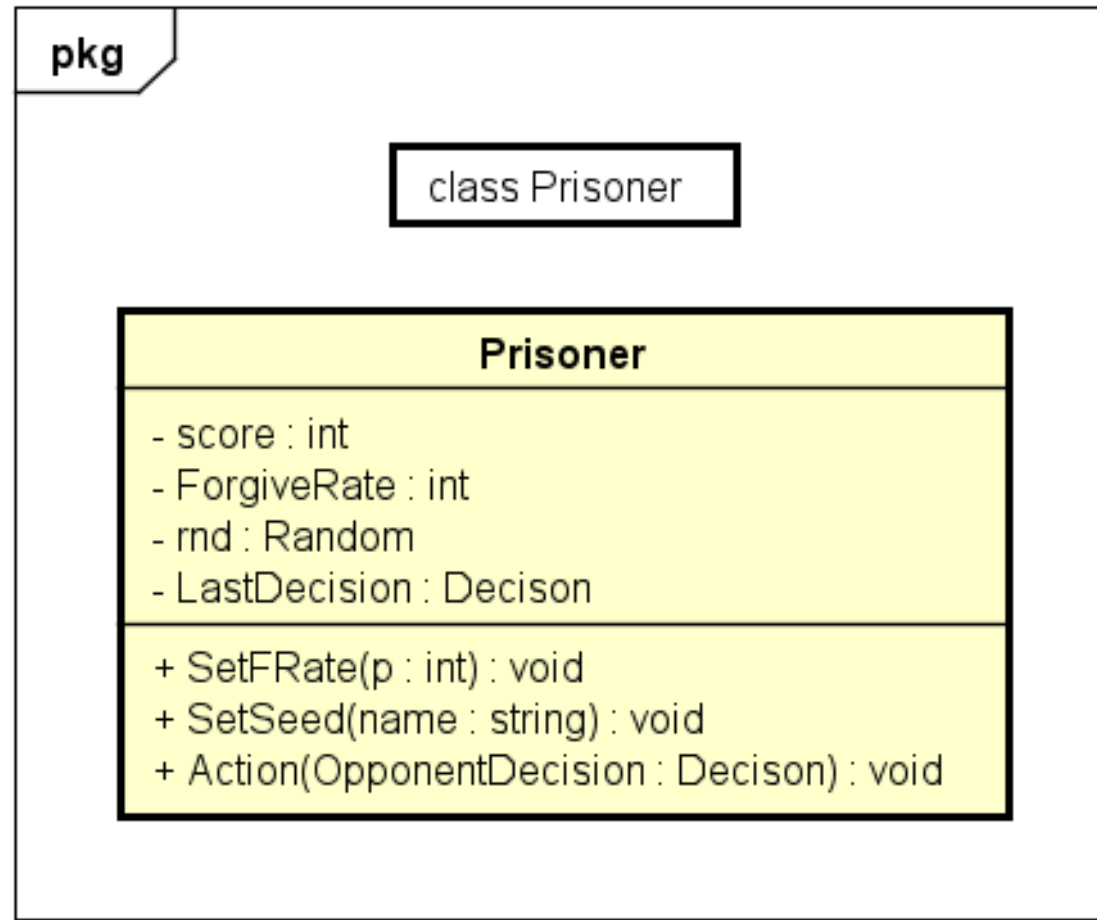
Action(Prisoner, Game) -

只跟Prisoner有關寫在Prisoner裡面，需要傳入對方Prisoner的LastDecison

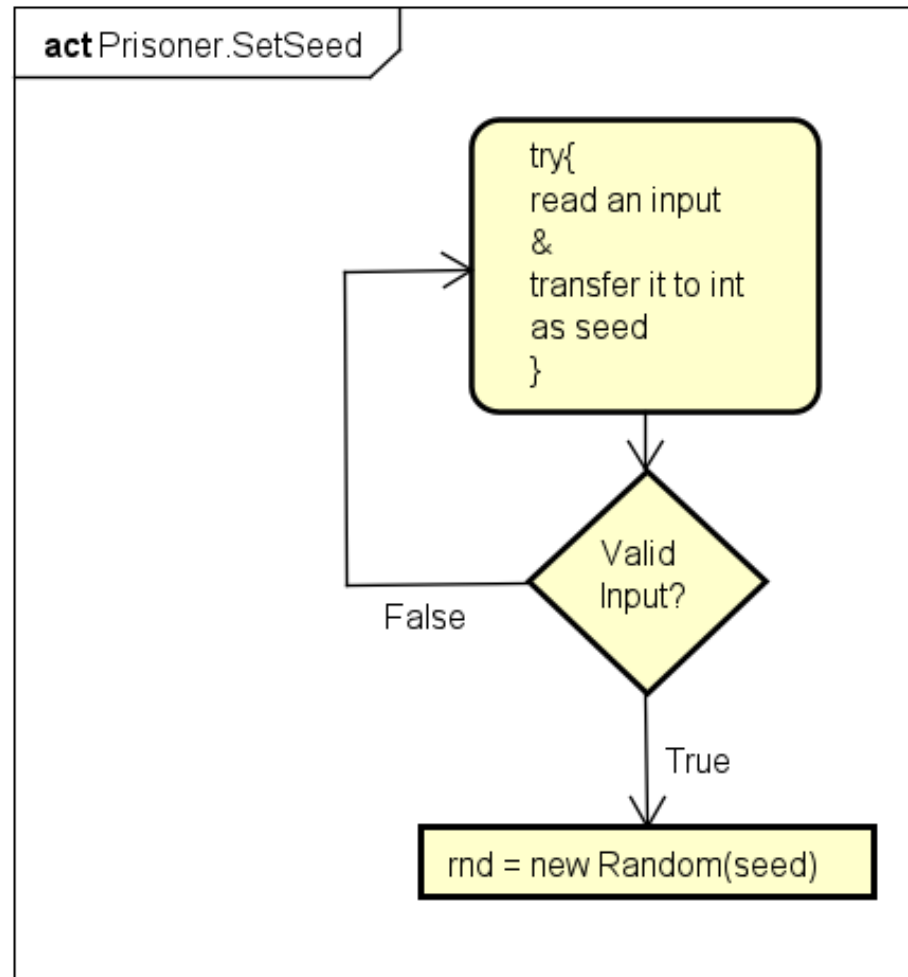
流程圖-Main



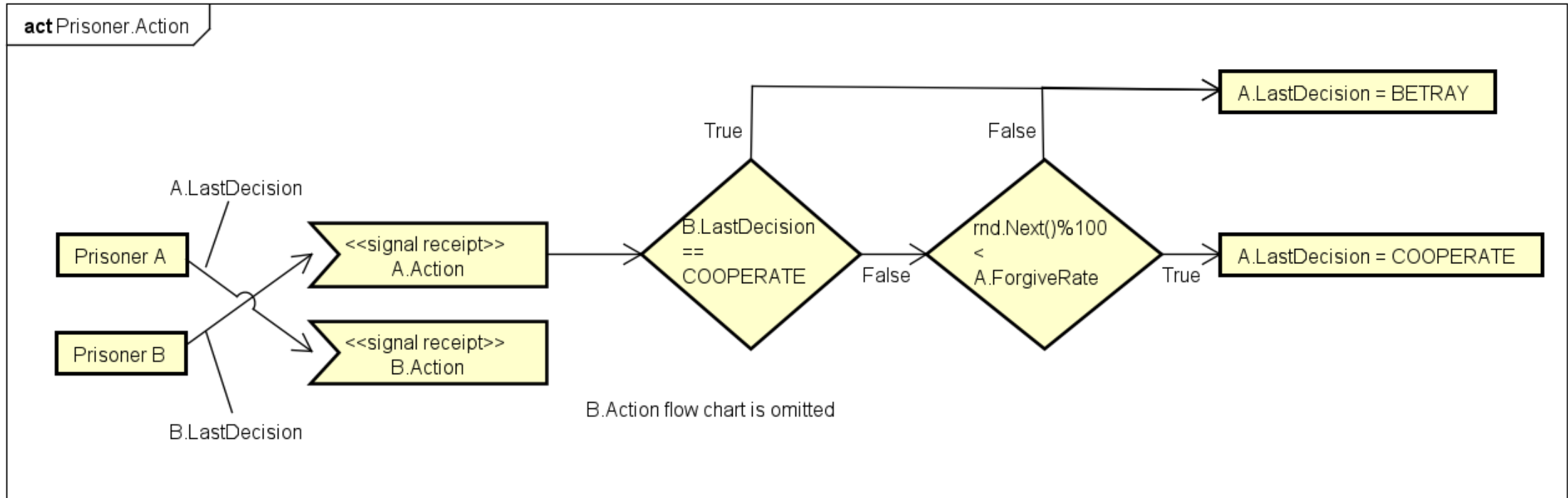
物件圖-Prisoner



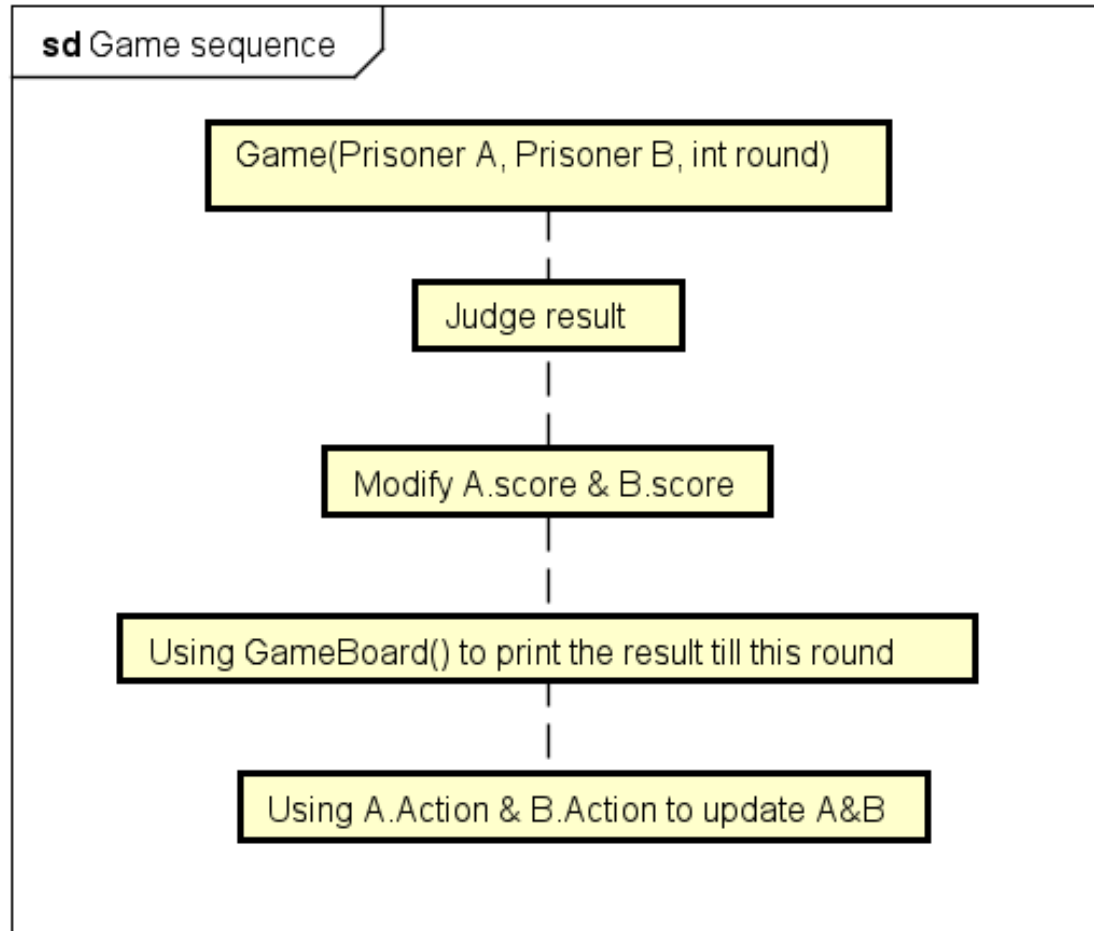
流程圖-SetSeed



流程圖-Action



流程圖-Game



結語

相信有人作業，是先寫出**Code**，後來才整理成流程規劃圖

這題我也是先寫完，才生出了很多廢話跟圖。

寫大程式時規劃必然重要，

但像這類小程式，

程式規劃圖可以幫助他人快速理解你的邏輯，跟這隻程式到底在幹嘛。

完整程式碼

► <https://reurl.cc/OXav5A>