

## Ponte del Diavolo (PDD) Game Requirements

### 1. Introduction

本文件簡述 Final Project 的需求，並說明整體 PDD 遊戲平台以及同學所設計撰寫的 PDD 策略程式在遊戲平台中的應有責任 (Responsibility)。由於同學彼此之間的策略將會互相進行對戰，所以進行策略實做時，必須遵守溝通介面的規定，以確保同學間的策略程式能互相溝通。本文件定義須遵守的溝通介面，請同學務必遵循溝通介面來完成策略程式的撰寫。

### 2. Final Project Requirements

請同學依照助教所規定之條件及溝通介面以 C Programming Language 進行 **PDD 遊戲策略程式的設計及實作**，並且能夠使用該策略程式成功地在助教所提供的平台上進行遊戲。同學所設計撰寫的遊戲策略程式的執行週期為一次的下棋決策，以下詳述該策略程式的需求。

1. 本次 Final Project 的 PDD 遊戲，遊戲平台會讓兩位玩家使用各自的策略程式進行遊戲
2. 該策略程式會預期從執行時的執行參數得知棋子分布狀況檔案 (chess.txt)、橋分布狀況檔案 (bridge.txt)、決策紀錄檔案 (stepLog.txt) 的路徑
3. 該策略程式會讀取棋子分布狀況檔案 (chess.txt)、橋分布狀況檔案 (bridge.txt) 的內容，得知目前的棋盤狀態
4. 該策略程式會讀取決策紀錄檔案 (stepLog.txt) 的內容，得知截至目前為止雙方玩家所有決策紀錄
5. 依據目前的棋盤狀態以及雙方玩家所有決策記錄，該策略程式會決定此次步驟要執行的動作內容
6. 該策略程式會根據此次步驟要執行的動作內容，來變動棋盤目前的狀態，並依格式規定將更改後的棋盤狀況覆蓋寫入至棋子分布狀況檔案 (chess.txt)、橋分布狀況檔案 (bridge.txt) 中，最後回傳 0 (return 0) 結束程式

註：策略程式不需要且不能更改 stepLog.txt 中的內容

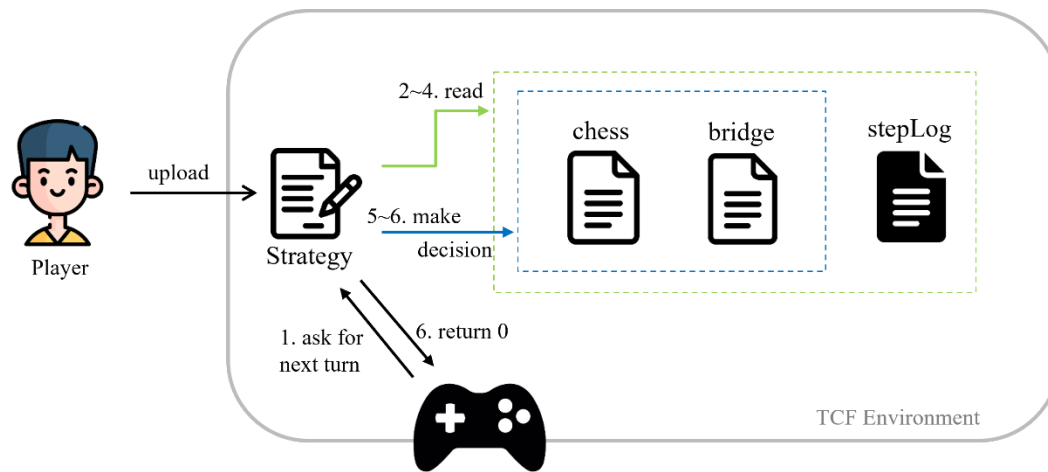
### 3. Game Rules

本節說明遊戲平台進行時的規則，同學的策略程式需依照 PDD 遊戲規則進行決策（PDD 遊戲規則請參考之前發佈的 PDD Game Introduction 文件），一旦有違規情形發生時，違規策略將被強制判輸，由敵方策略獲勝，違規狀況如下表所示。

Compile Error	Compile Error
選取的棋位已存在棋子	策略程式想要放置的棋位已存在棋子
選取棋位已被橋佔據	策略程式想要放置的棋位已被橋佔據
與島嶼對角相鄰	策略程式想要放置的棋子讓一座島嶼與其他同顏色之島嶼或沙洲對角相鄰
棋子相鄰數超過上限	策略程式想要放置的棋子讓五枚以上相同顏色的棋子相鄰
橋身所需佔據的空地存在棋子或其他橋	橋身經過的地方存在棋子或其他橋
橋的落點數有誤	策略程式在一枚棋子上建立第二座橋、橋的落點少於兩點
未使用合法棋位建立橋	策略程式建立橋所使用的兩個棋位不存在己方的棋子
不是規定的任一動作	策略程式進行的動作不是 Chess、Change、Bridge 其中之一
橋的距離錯誤	橋的兩個落點距離錯誤
超時	策略程式進行該回合動作的時間，超過 5 秒
竄改 stepLog.txt	策略程式對 stepLog.txt 進行非法寫入
寫檔格式錯誤	策略程式對 chess.txt 或 bridge.txt 寫檔之格式錯誤，例如建立新橋時編號錯誤或是非寫入 10*10 的棋盤狀況
顏色錯誤	棋子顏色與當前策略程式顏色不一致

表一、違規狀況

## 4. Overall Architecture



圖一、遊戲平台架構圖

上圖呈現整體的 PDD 遊戲平台的運作方式，由助教提供的遊戲平台會負責整個遊戲進行的控制、遊戲規則的仲裁以及遊戲畫面的呈現（此部分由助教提供），而 Strategy 則為學生所撰寫的遊戲策略程式。

在每個回合，遊戲平台會要求 Strategy 進行決策，並提供給 Strategy 最新的棋子分布狀況檔案 (chess.txt)、橋分布狀況檔案 (bridge.txt) 和決策紀錄檔案 (stepLog.txt)，Strategy 要依據這三份檔案所提供的資訊進行決策，並將執行該決策後的棋盤結果覆寫進棋子分布狀況檔案 (chess.txt) 以及橋分布狀況檔案 (bridge.txt) 當中，最後再回傳 0 表示已完成決策。

完成決策後，遊戲平台會檢視棋子分布狀況檔案 (chess.txt) 以及橋分布狀況檔案 (bridge.txt) 的內容，並決定是否已分出勝負或是繼續進行遊戲。

## 5. Program Controlling Interface

同學的程式編譯後，在 PDD 遊戲平台中，會用以下指令在助教提供的 Yellow Server 環境下，以 command line 執行程式

```
[學號].out [Color] [Y/N] [chess] [bridge] [stepLog]
```

- [Color]: 此策略程式所能操控的棋子顏色，資料值為 W 或 B，W 代表策略負責操控白方棋子，B 代表策略負責操控黑方棋子。
- [Y/N]: 此策略程式目前是否有權將第一回合白方放置的棋子變為黑色。資料值為 Y 或 N。

- `[chess]`: chess.txt 的檔案路徑，此檔案描述當前棋盤上棋子的分布狀況。
- `[bridge]`: bridge.txt 的檔案路徑，此檔案描述當前棋盤上橋的分布狀況。
- `[stepLog]`: stepLog.txt 的檔案路徑，此檔案紀錄了到目前為止雙方決策的歷史紀錄。

舉例來說

```
./E12345678.out B N ./chess.txt ./bridge.txt ./stepLog.txt
```

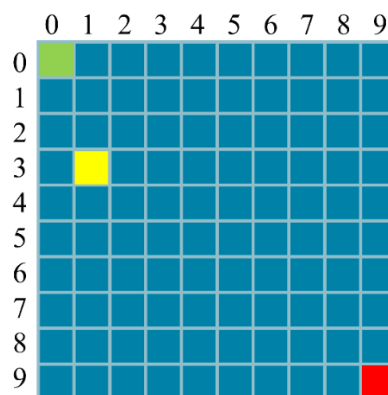
代表執行 E12345678 學生的策略程式進行一步棋步，其中

- 該策略程式控制的棋子為黑方的棋子
- 此策略程式無法將第一回合白方放置的棋子轉變為黑色。
- chess.txt 的檔案路徑為 ./chess.txt
- bridge.txt 的檔案路徑為 ./bridge.txt
- stepLog.txt 的檔案路徑為 ./stepLog.txt

## 6. 棋盤介紹

在遊戲進行時，遊戲平台會將棋子、橋呈現在棋盤中。

- 棋盤為圖二所示，棋盤以橫軸為 x 軸、縱軸為 y 軸，(x, y) 作為座標點，以圖中三個不同顏色的點作為範例：綠點座標為 (0, 0)、黃點座標為 (1, 3)、紅點座標為 (9, 9)



圖二、棋盤示意圖

## 7. Data Format

本節介紹遊戲進行時，遊戲平台會提供的三個檔案：chess.txt、bridge.txt 和 stepLog.txt，同學可依檔案所提供資訊，在策略程式中決定要進行什麼動作（下棋、建橋或轉變棋子顏色）。

### (1) 棋子分布狀況檔案格式 (chess.txt)

棋子分布狀況檔案紀錄了目前棋盤上棋子的分布狀況，排列成 10\*10 的陣列（不含逗點）。

下表是 chess.txt 各個值所代表的意義：

值	代表意義
1	白色棋子
2	黑色棋子
0	在棋盤上但無棋子的點

表二、chess.txt 數值代表含意

以下是棋子分布狀況檔案範例。

```

0  0,0,0,1,0,1,0,0,0,0
1  0,0,0,0,0,0,0,0,0,0
2  0,0,0,1,0,1,0,0,0,0
3  0,0,0,0,0,0,0,0,0,0
4  0,0,0,2,0,2,0,0,0,0
5  0,0,0,0,0,0,0,2,2,0
6  0,0,0,0,0,0,0,0,0,0
7  0,0,0,0,0,0,0,2,2,0
8  0,0,0,0,1,0,1,0,0,0
9  0,0,0,0,0,0,0,0,0,0

```

圖三、chess.txt 範例

**注意事項：**檔案當中每一行皆會以換行符號作為結束。

### (2) 橋分布狀況檔案格式 (bridge.txt)

橋分布狀況檔案紀錄了目前棋盤當中橋的分布狀況，每一座橋在棋盤上

都會有兩個落點，在 bridge.txt 檔案當中，每一座橋的兩個落點皆會由該座橋的編號做為標註，最新一座橋的編號會由當前棋盤上的橋數量決定，舉例來說現在棋盤上有 2 座橋，則策略程式想對 bridge.txt 寫入一座橋時，該座橋的編號就會是 3。

下表是 bridge.txt 各個值所代表的意義：

值	代表意義
1 以上	數字的值即該座橋的編號
0	在棋盤上但非橋的落點

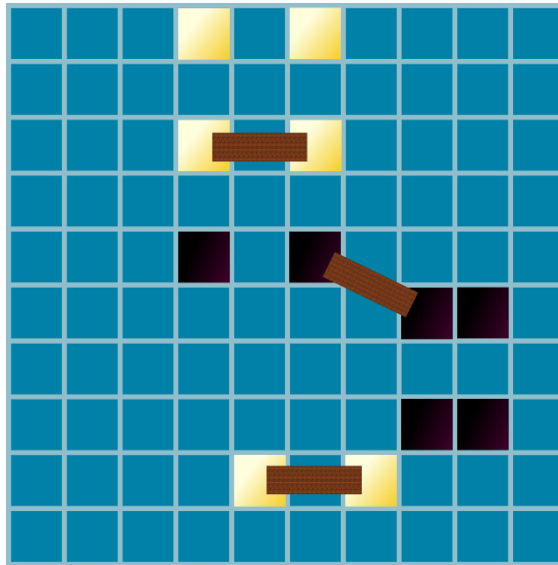
表三、bridge.txt 數值代表含意

以下是橋分布狀況檔案範例，此範例與圖三範例在遊戲中的棋盤畫面如圖五所示。

1	0,0,0,0,0,0,0,0,0,0
2	0,0,0,0,0,0,0,0,0,0
3	0,0,0,1,0,1,0,0,0,0
4	0,0,0,0,0,0,0,0,0,0
5	0,0,0,0,0,0,2,0,0,0
6	0,0,0,0,0,0,0,0,2,0
7	0,0,0,0,0,0,0,0,0,0
8	0,0,0,0,0,0,0,0,0,0
9	0,0,0,0,3,0,3,0,0,0
10	0,0,0,0,0,0,0,0,0,0

圖四、bridge.txt 範例

**注意事項：**檔案當中每一行皆會以換行符號作為結束。



圖五、棋盤範例

### (3) 決策紀錄檔案 (stepLog.txt)

決策紀錄檔案依序記錄了整個遊戲中策略程式進行的每一步動作，檔案中每行即代表一次策略程式的動作：

格式: [Color]\_[Turn]\_[Action]\_[Position\_1]\_[Position\_2]

- **Color**  
意義: 執行此動作的顏色  
資料值: W (白色) 或 B (黑色)
- **Turn**  
意義: 執行此動作的所屬回合，每次策略程式進行一步動作，回合數就會改變  
資料值: 1, 2, ...
- **Action**  
意義: 執行的動作為何  
資料值: Chess (下棋)、Bridge (建橋)、Change (黑方將第一回合白方下的兩枚棋子調換顏色)
- **Position\_1、Position\_2**  
意義: 棋子或橋放置的位置  
資料格式: x,y  
資料值: 如果 Action 是 Change，則 Position\_1、Position\_2 分別代表被調換顏色的兩枚棋子的位置；如果 Action 是 Chess，則 Position\_1、Position\_2 分別代表兩顆棋子放置的位置；如果 Action 是 Bridge，則 Position\_1、Position\_2 代表橋在棋盤上的兩個落點位置

以下是決策紀錄檔案範例

```
1 W_1_Chess_2,0_2,2
2 B_2_Change_2,0_2,2
3 W_3_Chess_0,2_1,3
4 B_4_Bridge_2,0_2,2
5 W_5_Chess_8,8_9,9
```

圖六、stepLog.txt 範例

**注意事項：**檔案當中每一行皆會以換行符號作為結束。

## 7. 實作注意事項

- 在執行策略程式時，遊戲資訊檔案不一定與策略程式置於同一工作目錄。為減輕同學判定遊戲資訊檔案位置之負擔，執行策略程式時，所傳入的檔案路徑參數將直接指定「檔案路徑」。
- 在決定移動的棋子後，策略程式只需要更新棋子分布狀況檔案 (chess.txt)、橋分布狀況檔案 (bridge.txt) 內容即可，**不需要更新決策紀錄檔案 (stepLog.txt)** (遊戲進行中，助教開發的遊戲平台會負責更新該檔案內容)。
- 特別注意事項：**每回合平台都會判斷輪替到的策略是否還有下棋或建橋的空間，如果無法再進行下棋或建橋的動作，平台會呼叫另一個策略程式進行決策；如果可以再下棋或建橋的話，平台會執行該策略，**當該策略未進行下棋或建橋的動作，平台會判定該策略違規**。因此同學實作策略時，需要特別判斷是否還有下棋或建橋的空間，以免被判違規。