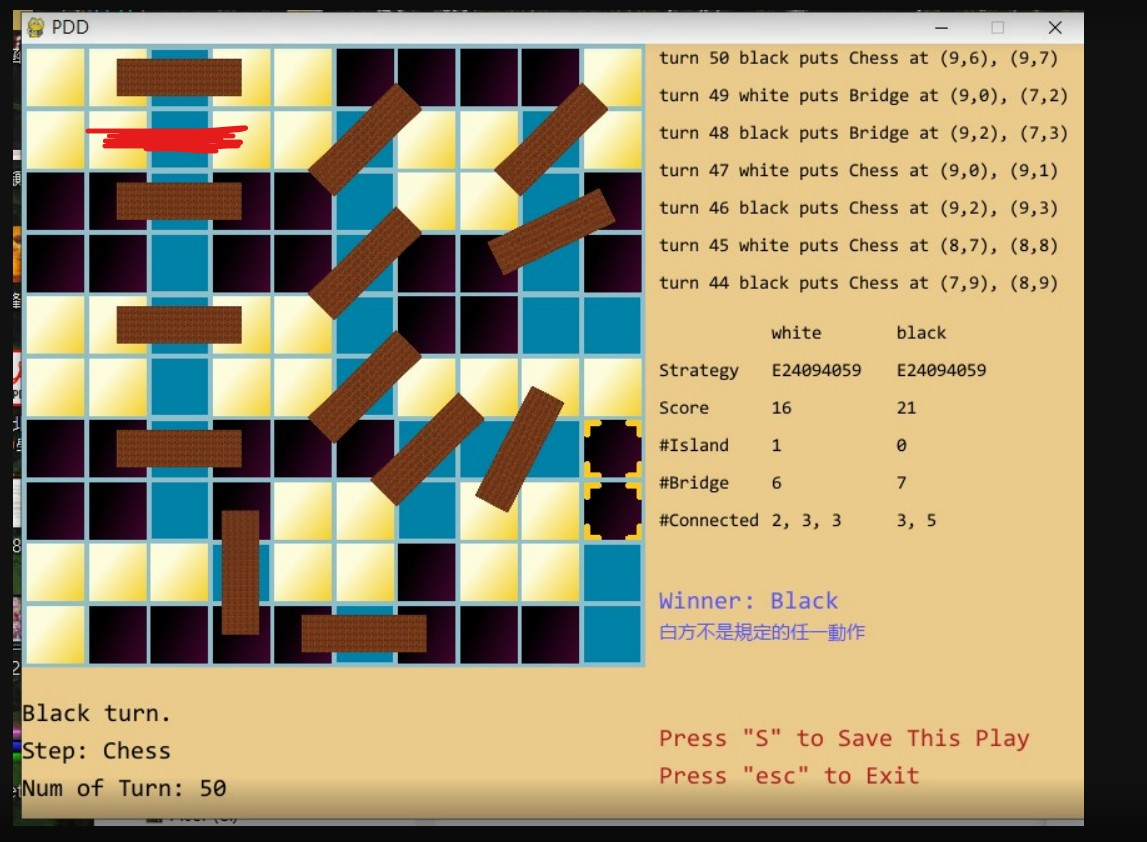
**策略邏輯：**

1. 資訊的使用：

此程式使用到了chess.txt、bridge.txt中的各個棋子和橋的分布情形，並沒有使用到stepLog.txt的內容。

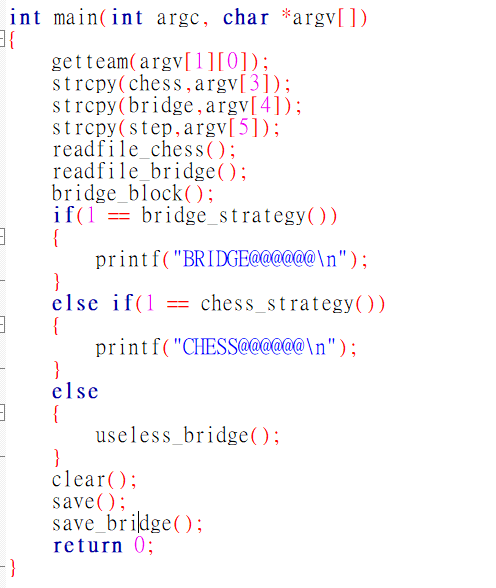
1. 策略的決定：

當可以建立橋時就建立，但是不重複在相同的島嶼之間建立橋，避免浪費步數，棋子的會是如過有2個相連則優先讓其形成島嶼，若沒有則另尋空間，主要仍然是將棋子相連做擺放，最後才是將剩餘的橋做擺放的動作(如圖一中的紅線部分)。

(圖一)

**策略架構：**

此程式將各個功能做function的區分(如圖二所示)

(圖二)

策略

讀檔

存檔

住要架構是先讀檔在策略最後存檔，策略則是優先蓋橋，後下棋，使用一般的二維array做分析檔案。使用了function，將各個功能分開可以使其他使用者或是程式開發，有效率地找到相關的功能並擴充或是優化。

下棋的策略是先將可以放的地方列出不能放的地方我是將其的值設為負數，並用各種可能出現的島嶼圖形做分割。

橋的策略則為將橋可能出現的情況作為基本的數據並尋找可能的位置，並將短的橋先放在放長的以必免出現橋交叉的情況。