* 策略邏輯：
  + 資訊的使用：此程式使用了chess.txt、bridge.txt，中的棋子和橋的位置作為程式判斷的依據。
  + 策略的決定：程式使用的策略是將步數最小化的方式將棋局下完，也就是盡量不要將步數使用在阻擋別人的方式，經營自己的島嶼，先將能夠相連的島嶼連，先將基本能拿的得分先得到，贏在起跑線上。
  + 策略選擇的原因和程式改進：因為我本身不太擅長桌遊，就是下跳棋或五子棋都很不太會贏的那種，所以在策略方面沒有太多的想法，因此想出來的策略也是比較普通的那種，不過在實測後的成效比自己所想的好一些，就保留目前的狀況，另外這次的程式和上次的沒有相差太多就是將一些沒有用到的內容刪掉，以優化程式，主要有兩個原因：一、我不太懂得要如何在棋盤上和對手勾心鬥角，所以選擇這種自我經營的策略，所以沒有做太大的更動。二、如同我在計概網站所問的，我的testbed好像有些問題無法開啟，也有試過助教所建議的方法仍然不能解決，在做debug時，就需要使用到家裡的電腦或是同學的，不過大家也有各自的事情要忙不可能無時無刻待在電腦前幫我，況且在上次的作業他們已經幫我很多了，所以只做程式上的簡化。
* 策略架構：

程式主要的架構是讀出檔案，存入array中，依照優先及執行策略：1能下橋就下橋2 下貼在一起的棋子3 下分開的棋子4 將剩餘的橋填滿，最後存檔。

優點：程式容易書寫，容易連出較多的島嶼，程式容易預測擺放位置比較不會出現程式和預期不一樣的現象。

缺點：當對手故意擋棋路時容易出現得分較低的情況，或是連線的只是沙洲。

程式的切割適用下棋和下橋做分割，先判斷可否下橋再來下棋。