

一、開啟新局

(1)編譯結果

```
sandra@LAPTOP-AD1R60F7:~/chess2$ gcc main.c
sandra@LAPTOP-AD1R60F7:~/chess2$
```

圖一。編譯結果。

(2)執行結果

引數: -n -s shogi.txt

如下圖一，玩家先輸入想移動的棋子的座標，再輸入該棋子的目的地，都先輸入行在輸入列。確認移動方式符合規則後，則會顯示新的棋盤，若不符合，則會請該玩家在輸入一次。移動完之後再輪到另一位玩家。

```
1 2 3 4 5 6 7 8 9
香桂銀金王金銀桂香1
 飛      角 2
步步步步步步步步3
      4
      5
      6
步步步步步步步步7
 角      飛 8
香桂銀金王金銀桂香9
player x enter the coordinate of the chess wh
ich you want to move:
2 7
player x enter the new coordinate of the ches
s:
2 6
1 2 3 4 5 6 7 8 9
香桂銀金王金銀桂香1
 飛      角 2
步步步步步步步步3
      4
      5
 步      6
步 步步步步步步7
 角      飛 8
香桂銀金王金銀桂香9
player y enter the coordinate of the chess wh
ich you want to move:

```

圖二。遊戲開始。

當某一方的「王」被吃掉時，則遊戲結束，並顯示哪一方贏得此棋局。如下圖三。

```

1 2 3 4 5 6 7 8 9
香桂銀金王金銀桂香1
  飛      角  2
步步步步步角步步3
      步      4
    步      5
  步      6
步步 步步步步步7
      飛  8
香桂銀金王金銀桂香9
player x enter the coordinate of the chess which
you want to move:
7 3
player x enter the new coordinate of the chess:
5 1
1 2 3 4 5 6 7 8 9
香桂銀金角金銀桂香1
  飛      角  2
步步步步步 步步3
      步      4
    步      5
  步      6
步步 步步步步步7
      飛  8
香桂銀金王金銀桂香9
Player x win!
sandra@LAPTOP-AD1R6OF7:~/chess2$

```

圖三。

玩家輸入 0 則可悔棋，棋盤便會顯示上一步驟的棋盤。當雙方還沒分出勝負時，玩家可輸入-1 結束遊戲。如下圖四。

```

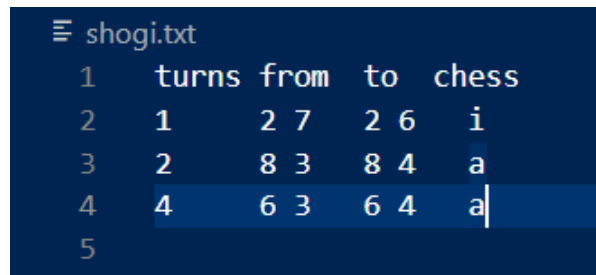
1 2 3 4 5 6 7 8 9
香桂銀金王金銀桂香1
  飛      角  2
步步步步步 步步3
      步      4
    步      5
  步      6
步步 步步步步步7
  角      飛  8
香桂銀金王金銀桂香9
player x enter the coordinate of the chess which you w
ant to move:
0
1 2 3 4 5 6 7 8 9
香桂銀金王金銀桂香1
  飛      角  2
步步步步步步步步3
      步      4
    步      5
  步      6
步步 步步步步步7
  角      飛  8
香桂銀金王金銀桂香9
player y enter the coordinate of the chess which you w
ant to move:
-1
End
sandra@LAPTOP-AD1R6OF7:/mnt/c/Users/asus/Desktop/NCU/
大二/datastructure/chess$

```

圖四。

下棋過程中也會將資料儲存到文字檔(shogi.txt)。檔案會儲存:回合數(turns)、移動的棋子代號(chess)、原本座標(from)以及目標座標(to)，座標儲存形式為:(行, 列)。如圖四。

如果玩家悔棋，則會直接蓋掉上一輪下棋所儲存的資料。因此圖四的回和數會從 2 跳到 4，是因為中間有悔棋的過程。

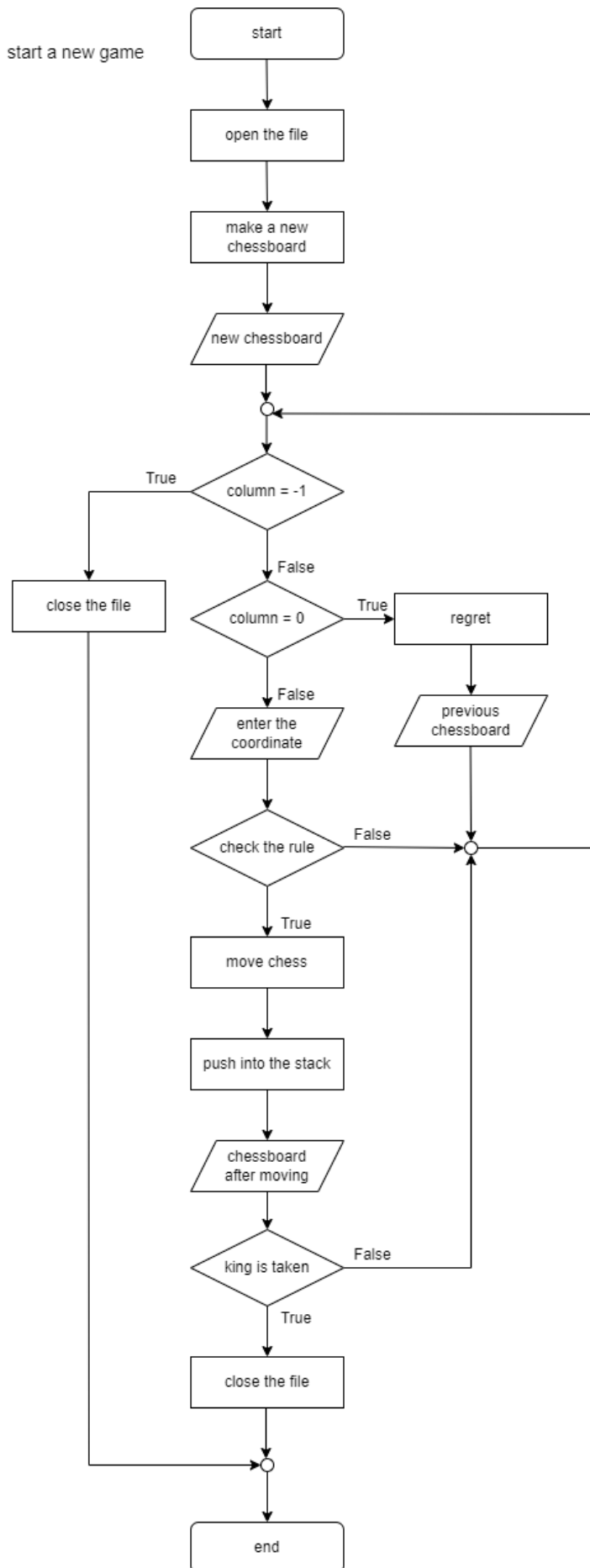


	turns	from	to	chess
1	1	2 7	2 6	i
2	2	8 3	8 4	a
3	4	6 3	6 4	a
4				
5				

圖五。

(3)流程圖(見下頁)

start a new game



二、讀取棋譜

(1) 編譯結果

```
sandra@LAPTOP-AD1R60F7:~/chess2$ gcc main.c -lm
sandra@LAPTOP-AD1R60F7:~/chess2$
```

圖六。編譯結果。

(2) 執行結果

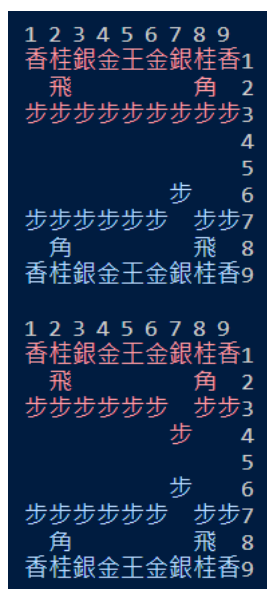
引數:-l shogi.txt

讀取之前所儲存的檔案，將資料存入 stack 裡，再依照 turns 的順序一一顯示出每個回合的棋盤。如圖八、圖九。

```
≡ shogi.txt
```

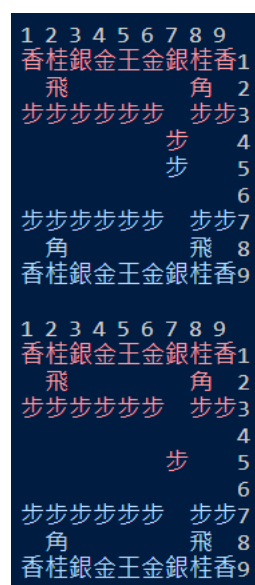
1	turns	from	to	chess
2	1	7 7	7 6	i
3	2	7 3	7 4	a
4	3	7 6	7 5	i
5	4	7 4	7 5	a
6				

圖七。此圖為前一局下棋時所存的檔案。



1	2	3	4	5	6	7	8	9
香	桂	銀	金	王	銀	桂	香	1
飛						角		2
步	步	步	步	步	步	步	步	3
							步	4
					步			5
步	步	步	步	步	步	步	步	6
角						飛		7
香	桂	銀	金	王	銀	桂	香	8
香	桂	銀	金	王	銀	桂	香	9

圖八。



1	2	3	4	5	6	7	8	9
香	桂	銀	金	王	銀	桂	香	1
飛						角		2
步	步	步	步	步	步	步	步	3
							步	4
					步			5
步	步	步	步	步	步	步	步	6
角						飛		7
香	桂	銀	金	王	銀	桂	香	8
香	桂	銀	金	王	銀	桂	香	9

圖九。

(3) 流程圖

read a old game

