

Assignment #2

110503523 通訊二 王以豪

1. 編譯結果：

```
● richie0203@RichieG15:~/workspace/ShogiPtr$ make
gcc -c src/rule.c -o build/rule.o
gcc -c src/main.c -o build/main.o
gcc -c src/data.c -o build/data.o
gcc -c src/watcher.c -o build/watcher.o -lev
gcc build/rule.o build/main.o build/data.o build/watcher.o -o bin/chess -lev
```

2. 執行結果：

```
○ richie0203@RichieG15:~/workspace/ShogiPtr$ make run
```

```
上方玩家：0:10
下方玩家：0:12

  9  8  7  6  5  4  3  2  1
香 桂 銀 金 王 金 銀 桂 ． 一
． 飛 ． ． ． ． ． 角 ． 二
步 步 ． 步 步 步 步 步 香 三
． ． 步 ． ． ． ． ． 步 四
． ． ． ． ． ． ． ． ． 五
． ． ． 步 ． ． ． ． 步 六
步 步 步 ． 步 步 步 步 ． 七
． 角 ． ． ． ． ． 飛 ． 八
香 桂 銀 金 玉 金 銀 桂 香 九

- 輪到下方玩家 -
輸入目標的座標x與y:
> 
```

```
richie0203@RichieG15:~/workspace/ShogiPtr$ make load
```

```
步
9 8 7 6 5 4 3 2 1
香 桂 銀 金 王 ． 銀 桂 ． 一
． 飛 ． ． 金 ． ． 角 ． 二
步 步 ． 步 步 步 步 ． ． 三
． ． 步 ． ． ． ． 步 ． 四
． ． ． ． ． ． ． ． 香 五
． ． ． 步 ． ． 步 ． ． 六
步 步 步 ． 步 步 ． 步 ． 七
． 角 ． ． ． ． ． 飛 ． 八
香 桂 銀 金 玉 金 銀 桂 ． 九
步 香
輸入 'f' 移動下一手 'b' 移動上一手
    'p' 遊玩模式    'e' 退出
> 
```

3. 功能介紹：

(1) 按下 E/e 將儲存遊戲並終止遊戲。

```
上方玩家：0:58
下方玩家：0:20
步
9 8 7 6 5 4 3 2 1
香 桂 銀 金 王 ． 銀 桂 ． 一
． 飛 ． ． 金 ． ． 馬 ． 二
步 步 ． 步 步 步 ． ． ． 三
． ． ． ． ． ． 步 步 ． 四
． ． 步 步 ． ． ． ． 香 五
． ． 步 ． ． ． 步 ． ． 六
步 步 ． ． 步 步 ． 步 ． 七
． ． ． ． ． ． ． 飛 ． 八
香 桂 銀 金 玉 金 銀 桂 ． 九
步 香 角
- 輪到上方玩家 -
輸入目標的座標x與y:
> e
Save and exit the game!
```

(2) 按下 S/s 將儲存遊戲(可繼續執行)

```
上方玩家: 0:07
下方玩家: 0:45

 9  8  7  6  5  4  3  2  1
香 桂 銀 金 王 金 銀 桂 香 一
· 飛 · · · · · 角 · 二
步 · 步 步 · 步 步 步 · 三
· 步 · · 步 · · · · 四
· · · 步 · · · · 步 五
· · · · · 步 · · 步 六
步 步 步 · 步 · 步 步 · 七
· 角 · · · · · 飛 · 八
香 桂 銀 金 王 金 銀 桂 香 九

- 輪到上方玩家 -
輸入目標的座標x與y:
> s
Game save!
```

4. 分析：

遇到的問題：

(1) (已解決)目前在讀取棋盤模式時，雖能正常運作。然而從該模式輸入 P 進入遊玩模式時，雖已可以將新的數據覆蓋到原數據上，但無法將沒覆蓋的後續數據移除。

- 目前解法：在寫檔前偵測 maxcount(該次讀檔最大手數)是否大於 count(當前回合手數)，若成立則在寫檔時多寫進一個"-1"，下次讀檔如果遇到"-1"便直接 break 跳離迴圈並將 maxcount 更新。

```
≡ map.txt
 1  4 2 B 4 3 t
 2  4 6 b 4 5 t
 3  4 3 B 4 4 t
 4  4 5 b 4 4 B
 5  8 2 B 8 3 t
 6  4 4 b 4 3 t
 7  8 3 B 8 4 t
 8  4 3 b 4 2 t
 9  8 4 B 8 5 t
10  4 2 b 4 1 t
11  8 5 B 8 6 b
12
```

原檔案(選擇的座標/棋子名稱/前往的座標/前往的位置名稱)

```
上方玩家: 0:00
下方玩家: 0:00

9 8 7 6 5 4 3 2 1
香 桂 銀 金 王 金 銀 桂 香 一
· 飛 · · · · 角 · 二
步 步 步 步 · 步 步 步 步 三
· · · · · · · · · 四
· · · · 步 · · · · 五
· · · · 步 · · · · 六
步 步 步 步 · 步 步 步 步 七
· 角 · · · · · 飛 · 八
香 桂 銀 金 王 金 銀 桂 香 九

- 輪到下方玩家 -
輸入目標的座標X與Y:
> |
```

移動上一手直到第3手時

```
4 2 B 4 3 t
4 6 b 4 5 t
4 3 B 4 4 t
4 5 b 4 4 B
8 2 B 8 3 t
4 4 b 4 3 t
8 3 B 8 4 t
```

應從紅線處下方開始續寫

```
4 2 B 4 3 t
4 6 b 4 5 t
4 3 B 4 4 t
3 8 j 4 7 t
8 0 S 8 1 t
4 4 b 4 3 t
3 3 B 8 4 t
```

做了2次移動後(移動 j 及 S) 理論上遊戲只進行到紅框以上，但紅框(及以下)的棋譜並未移除，將導致下一次讀檔產生錯誤。

| | | | | | | |
|----|----|---|---|---|---|---|
| 1 | 4 | 2 | B | 4 | 3 | t |
| 2 | 4 | 6 | b | 4 | 5 | t |
| 3 | 4 | 3 | B | 4 | 4 | t |
| 4 | 5 | 8 | j | 4 | 7 | t |
| 5 | 0 | 0 | S | 0 | 1 | t |
| 6 | -1 | | | | | |
| 7 | | b | 4 | 3 | t | |
| 8 | 8 | 3 | B | 8 | 4 | t |
| 9 | 4 | 3 | b | 4 | 2 | t |
| 10 | 8 | 4 | B | 8 | 5 | t |
| 11 | 4 | 2 | b | 4 | 1 | t |
| 12 | 8 | 5 | B | 8 | 6 | b |
| 13 | | | | | | |

目前解決方法 → 寫入「-1」進檔案中

(2) (部份解決)升變的棋子在吃掉後被對方打入，在悔棋及復盤方面(chessBack)會顯示錯誤。

- 目前解法：將升變後被打入的棋子以另外的符號紀錄，在被回復時(進入駒台時)將變回升變後的符號。
- 尚未解決：升變過後的棋子在

```
/* Change the chess that already promoted and being dropped*/
void changeChess(char *c){
    switch(*c){
        case 'U': //U -> 馬
            *c = 'H';
            break;
        case 'L': //L -> 龍
            *c = 'D';
            break;
        case 'N': //N -> 全
            *c = 'A';
            break;
        case 'Z': //Z -> 杏
            *c = 'X';
            break;
        case 'I': //I -> 圭
            *c = 'P';
            break;
        case 'Q': //Q -> と
            *c = 'K';
            break;
    }
}
```

目前解決方法 → 原先升變後的馬「H」變成「U」。在要被回復(chessBack)時執行 changeChess 函式將棋子變回來並放進駒台