## Assignment\_2

## 1. 編譯結果

casper@LAPTOP-1DKFTNB9:~/work/Shogi\$ make

```
gcc -c src/main.c -o build/main.o
gcc -c src/shogi.c -o build/shogi.o
gcc -c src/file.c -o build/file.o
gcc -c src/timer.c -o build/timer.o
gcc build/main.o build/shogi.o build/file.o build/timer.o -o bin/shogi -lev
```

## 2. 執行結果

casper@LAPTOP-1DKFTNB9:~/work/Shogi\$ ./bin/shogi -n -s chessgame1.txt

玩家x遊戲時間:00:00:01 玩家y遊戲時間:00:00:00

9 8 7 6 5 4 3 2 1

\_\_\_\_\_

打入棋子請輸入H 悔棋請輸入0 儲存先前對弈過程請輸入S 關閉遊戲請輸入X

玩家X輸入移動棋子的 X 座標與 Y 座標: ■

玩家x遊戲時間:00:00:13 玩家x遊戲時間:00:00:13 玩家y遊戲時間:00:00:06 玩家y遊戲時間:00:00:06 987654321 987654321 \_\_\_\_\_ \_\_\_\_\_ ||香桂銀金王金銀・・||1 ||香桂銀金王金銀・・||1 ||・飛・・・・角・||2 |・飛・・・・角・||2 |步步步步步步步香||3  $||\cdot\cdot\cdot\cdot\cdot\cdot||_4$ | . . . . . . . . | | 5 | . . . . . . . . | |6 || . . . . . . . . . ||6 ||步步步步步步步。||7 ||・角・・・・飛・||8 ||・角・・・・飛・||8 ||香桂銀金玉金銀桂・||9 ||香桂銀金玉金銀桂・||9 打入棋子請輸入H 升變請輸入y 悔棋請輸入0 不升變請輸入n 儲存先前對弈過程請輸入S 關閉遊戲請輸入x 玩家X輸入移動棋子的 X 座標與 Y 座標: ■ 玩家x遊戲時間:00:00:17 玩家x遊戲時間:00:00:35 玩家y遊戲時間:00:00:20 玩家y遊戲時間:00:00:38 9 8 7 6 5 4 3 2 1 987654321 \_\_\_\_\_ \_\_\_\_\_ ||香桂銀金王金銀・・||1 ||香桂銀金王金銀・歩||1 ||・飛・・・・角・||2 ||・飛・・・・角・||2 |步步步步步步步查||3 ||步步步步步步步杳||3 || · · · · · · · ||4 ||・角・・・・飛・||8 ||・角・・・・飛・||8 ||香桂銀金玉金銀桂・||9 ||香桂銀金玉金銀桂・||9 打入棋子請輸入H 打入棋子請輸入H 悔棋請輸入0 悔棋請輸入0 儲存先前對弈過程請輸入S 儲存先前對弈過程請輸入S 關閉遊戲請輸入x 關閉遊戲請輸入x 玩家Y輸入移動棋子的 X 座標與 Y 座標: ■ 玩家X輸入移動棋子的 X 座標與 Y 座標: h請輸入要打入吃掉的第幾個棋子

請輸入要打入的位置

1 1

玩家x遊戲時間:00:01:21 玩家y遊戲時間:00:01:12

步

987654321

步王

● 遊戲結束!玩家2獲勝casper@LAPTOP-1DKFTNB9:~/work/Shogi\$

# 3. 分析

#### 遇到問題:

退回上一手請輸入r

繼續玩請輸入p

關檔請輸入x

- (1). 若在悔棋時存檔,已經儲存到檔案中的棋局無法刪除。
- (2). 讀檔時也會\n 也會儲存到字串中導致無法正常顯示棋局。
- (3). 在 head file 中宣告的全域變數,無法在 source file 中 include 進去
- (4). head file 與 source file 不同資料夾,編譯時找不到標頭檔無法編譯。

退回上一手請輸入r

繼續玩請輸入p

關檔請輸入x

(5). 無法將編譯完的輸出產生於指定資料夾

## 解決方法:

(1). 若偵測到悔棋時,則在檔案中寫 0,讀檔時讀到 0 結束,忽略以下 資料。

(2). 用 fscanf 直接讀\n 不存取以忽略它。

```
fscanf(cfptr, "%s", data[k].eatenchessy);
fscanf(cfptr,"\n");
for(j=0;j<ROW;j++){
    for(i=0;i<COLUMN;i++){
        fscanf(cfptr,"%c",&data[k].chess[j][i]);</pre>
```

(3). 在需要用到這些變數的一個 source file 中宣告,將 head file 中的宣告加上 extern 就可以正常 include 到其他 source 中了

```
C file.c
        M \times
                C main.c M
                                 C shog
src > C file.c > ...
  #include "../inc/file.h"
  2 #include"../inc/shogi.h"
  3
  4
       FILE *cfptr;
  5
C file.h
inc > C file.h > ...
  1 #ifndef FILE H
  2
       #define FILE H
  3
  4 #include "shogi.h"
  5
       extern FILE *cfptr;
```

(4). 將標頭檔的路徑寫在 include 裡面

```
#include"../inc/shogi.h"
#include"../inc/file.h"
```

(5). 加入指令 -o <file name> 可將輸出放置於<file name>

```
main.o: src/main.c
gcc -c src/main.c -o build/main.o
```

- 4. 参考資料
  - (1). <a href="https://shogi.hk/Gameplay-of-Japanese-Chess-Shogi/">https://shogi.hk/Gameplay-of-Japanese-Chess-Shogi/</a>
  - (2). https://www.796t.com/article.php?id=190246
  - (3). Example of Getopt (The GNU C Library)
  - (4). (2) Makefile 的写法 YouTube