Assignment 2 110503512 林妤蓁

一、編譯結果

```
sandra@LAPTOP-AD1R6OF7:/mnt/c/Users/asus/Desktop/NCU/
大二/datastructure/chess$ gcc main.c
sandra@LAPTOP-AD1R6OF7:/mnt/c/Users/asus/Desktop/NCU/
大二/datastructure/chess$
```

圖一、編譯結果

二、執行結果

玩家輸入 0 時可悔棋，棋盤便會回到上一個棋盤。若雙方還沒分出勝負時，玩家可輸入-1 結束遊戲。如下圖二。

```
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
                 5
    步           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 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
步步步步步步步步步3
                 4
                 5
    步           6
步步  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$
```

圖二

遊戲過程中，玩家每走一步會將下棋過程的資料(包含起始位置、目標位置、以及吃掉的棋子)存到以 linked list 串接的 stack 裡，悔棋時便可將資料讀取出來，顯示出前一步的棋盤。

```c
struct node{
    char start_chess;
    char end_chess;
    char eat;//被吃的棋子
    int start[2];//起點座標
    int end[2];//終點座標
    int eaten_chess[2];//被吃棋子的座標
    struct node *next;
}*stack=NULL;
```

圖三、stack 內容

```c
void push(int start1,int start2,int end1,int end2,int eat1,int eat2)
{
    NODE *temp;
    temp=(NODE*)malloc(sizeof(NODE));
    temp->start[0]=start1;
    temp->start[1]=start2;
    temp->end[0]=end1;
    temp->end[1]=end2;
    temp->eaten_chess[0]=eat1;
    temp->eaten_chess[1]=eat2;
    temp->eat=board[eat1][eat2];
    temp->start_chess=board[start1][start2];
    temp->end_chess=board[end1][end2];
    temp->next=stack;
    stack=temp;
}
```

圖四、將資料存入 stack，並將每一步棋的資料串連起來。

```c
int pop()
{
    if(stack==NULL){
        printf("Cannot go back anymore.\n");
        return 0;
    }
    else
    {
        NODE *temp;
        temp=stack;
        board[temp->start[0]][temp->start[1]]=temp->start_chess;
        board[temp->end[0]][temp->end[1]]=temp->end_chess;
        stack=stack->next;
        free(temp);
        turn++;
        return 1;
    }
}
```

圖五、悔棋時會取出 stack 中最新的資料，讓棋盤回到上一步

下棋過程中也會將資料儲存到文字檔。檔案會儲存:回合數(turns)、移動的旗子代號(chess)、原本座標(from)以及目標座標(to)。如圖六、圖七。

shogi.txt

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

圖六

```c
int main()
{
    if((new_game=fopen("shogi.txt","w+"))!=NULL)
        fprintf(new_game,"%-6s%-6s%-4s%-6s\n","turns","from","to","chess");
```

圖七、開檔

```c
else if(board[new_row][new_column]==0)//move chess
{
    push(origin_row,origin_column,new_row,new_column,new_row,new_column);
    fprintf(new_game,"%-6d%-2d%-2d%3d%2d%4c\n",turn,origin_column+1,origin_row,new_column+1,new_row,board[origin_row][origin_column]);
    board[new_row][new_column]=board[origin_row][origin_column];
    board[origin_row][origin_column]=0;
    turn++;
}
```

圖八、寫檔