Project Report for Data Structure

Simple Shogi

using linked-list

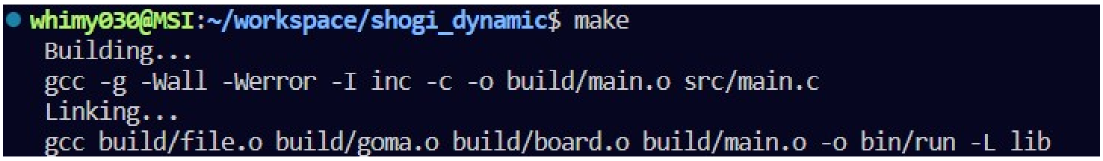
錢宇倫
通訊二
110503511

Department of Communication Engineering,

December 23, 2022

Contents

1 Makefile

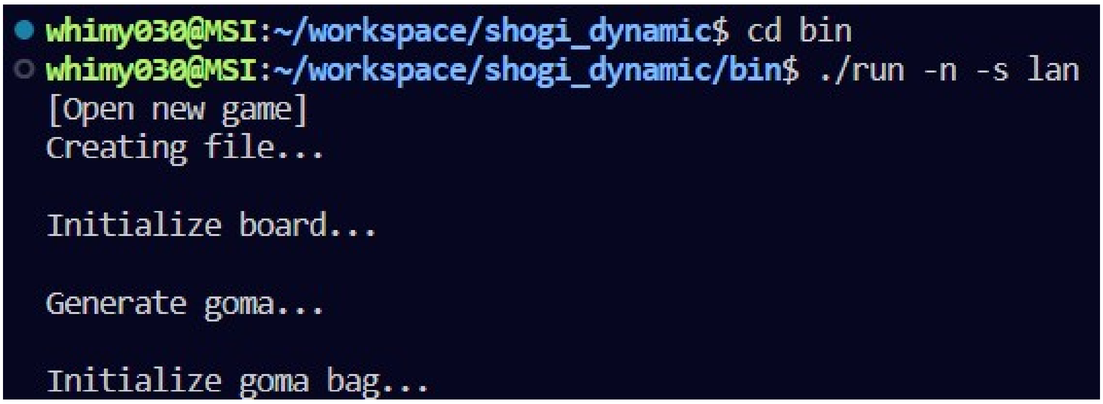


```
● whimy030@MSI:~/workspace/shogi_dynamic$ make
  Building...
  gcc -g -Wall -Werror -I inc -c -o build/main.o src/main.c
  Linking...
  gcc build/file.o build/goma.o build/board.o build/main.o -o bin/run -L lib
```

figure.1 - makefile

2 Open game

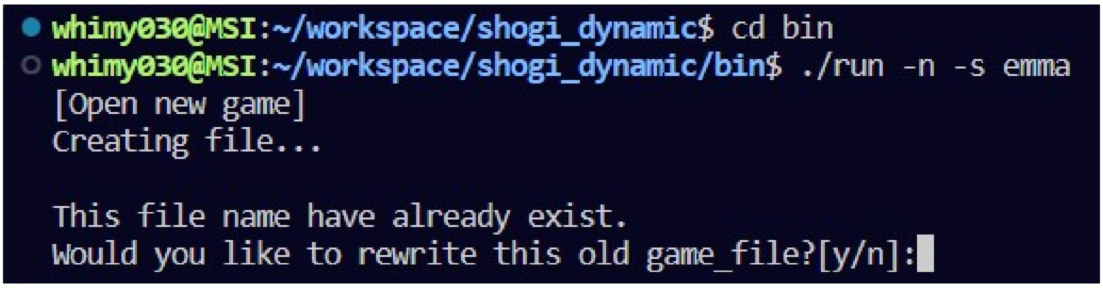
Open a new game...



```
● whimy030@MSI:~/workspace/shogi_dynamic$ cd bin
○ whimy030@MSI:~/workspace/shogi_dynamic/bin$ ./run -n -s lan
  [Open new game]
  Creating file...

  Initialize board...

  Generate goma...

  Initialize goma bag...
```

figure.2 - open new game

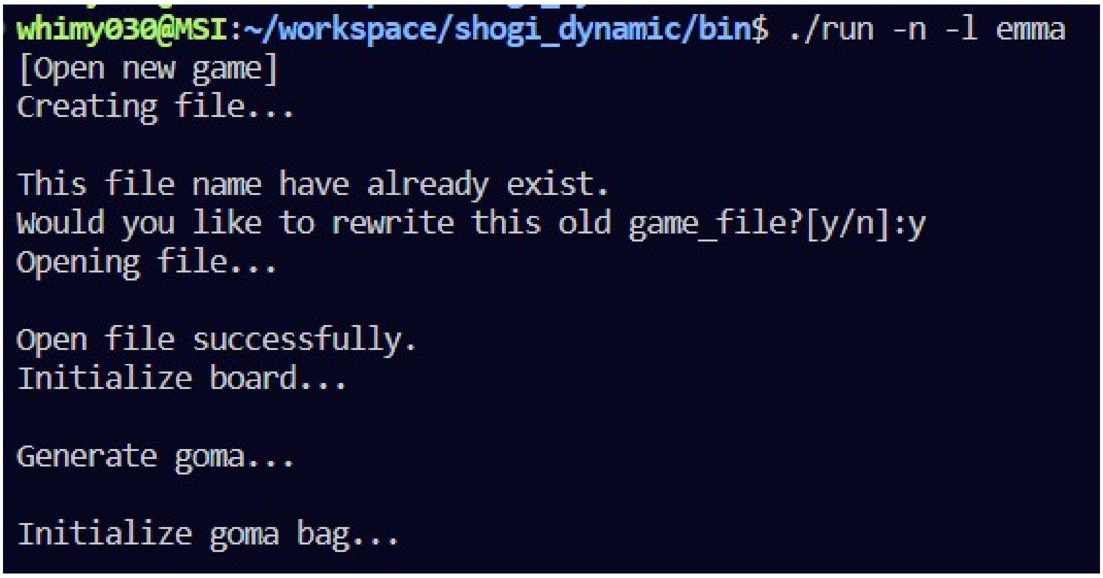
Open an existing game...



```
● whimy030@MSI:~/workspace/shogi_dynamic$ cd bin
○ whimy030@MSI:~/workspace/shogi_dynamic/bin$ ./run -n -s emma
  [Open new game]
  Creating file...

  This file name have already exist.
  Would you like to rewrite this old game_file?[y/n]:
```

figure.3 - open an existing game

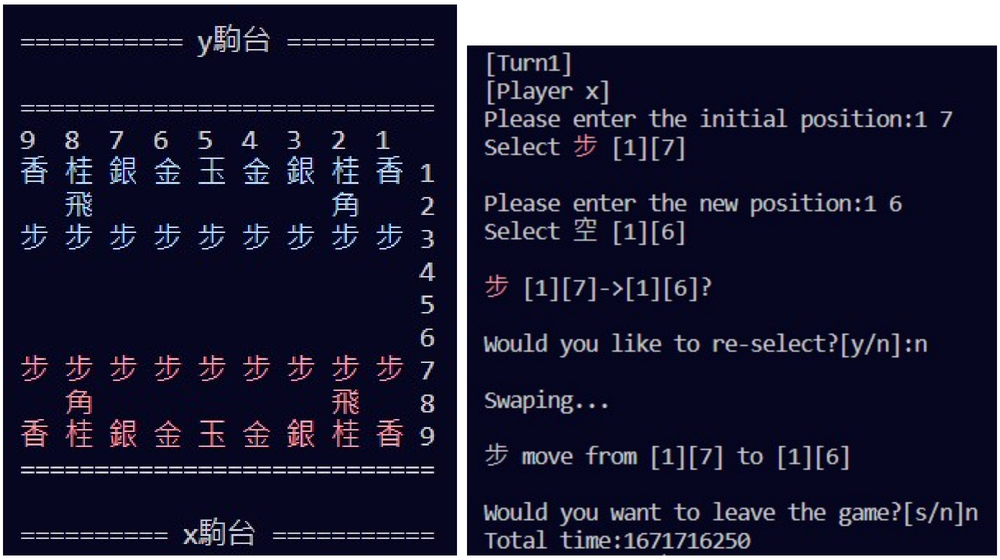
Cover an existing game...



```
whimy030@MSI:~/workspace/shogi_dynamic/bin$ ./run -n -l emma
[Open new game]
Creating file...

This file name have already exist.
Would you like to rewrite this old game_file?[y/n]:y
Opening file...

Open file successfully.
Initialize board...

Generate goma...

Initialize goma bag...
```

figure.4 - Cover an existing game

Basic display



```
=========== y駒台 ==========

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

========== x駒台 ===========
```

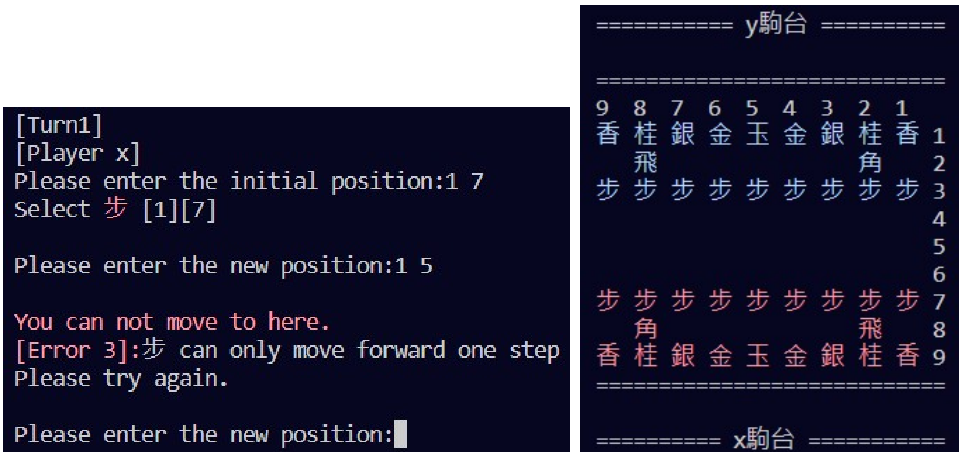
```
[Turn1]
[Player x]
Please enter the initial position:1 7
Select 步 [1][7]

Please enter the new position:1 6
Select 空 [1][6]

步 [1][7]->[1][6]?

Would you like to re-select?[y/n]:n

Swaping...

步 move from [1][7] to [1][6]

Would you want to leave the game?[s/n]n
Total time:1671716250
```

figure.5(a)(b) - (a)Board display(b)Basic move instruction

Error

- **Movement restriction->** `Not following the movement rule`

1: *Pawn* can only go on step forward each turn

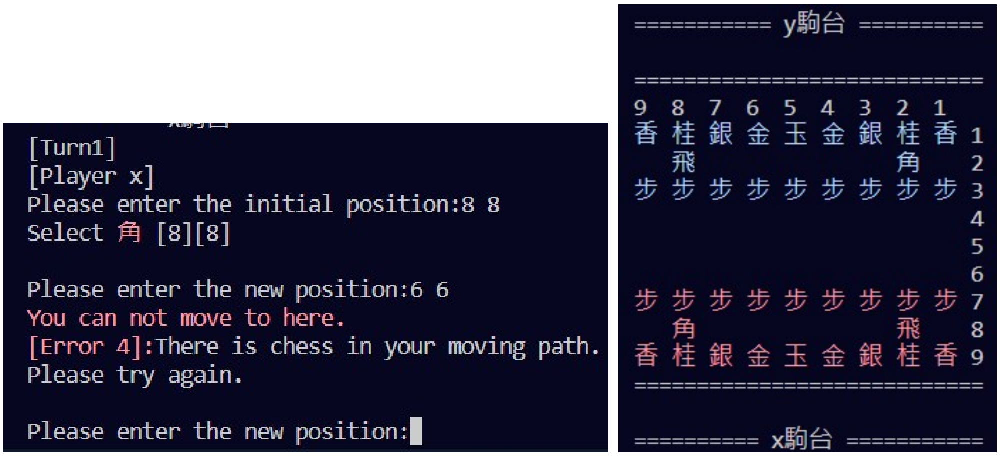


```
[Turn1]
[Player x]
Please enter the initial position:1 7
Select 步 [1][7]

Please enter the new position:1 5

You can not move to here.
[Error 3]:步 can only move forward one step
Please try again.

Please enter the new position:
```

```
=========== y駒台 ==========

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

========== x駒台 ===========
```

figure.6(a)(b) - (a)Error message(*Pawn* move)(b)example board

Case 2: *Bishop* can only move in a diagonal direction

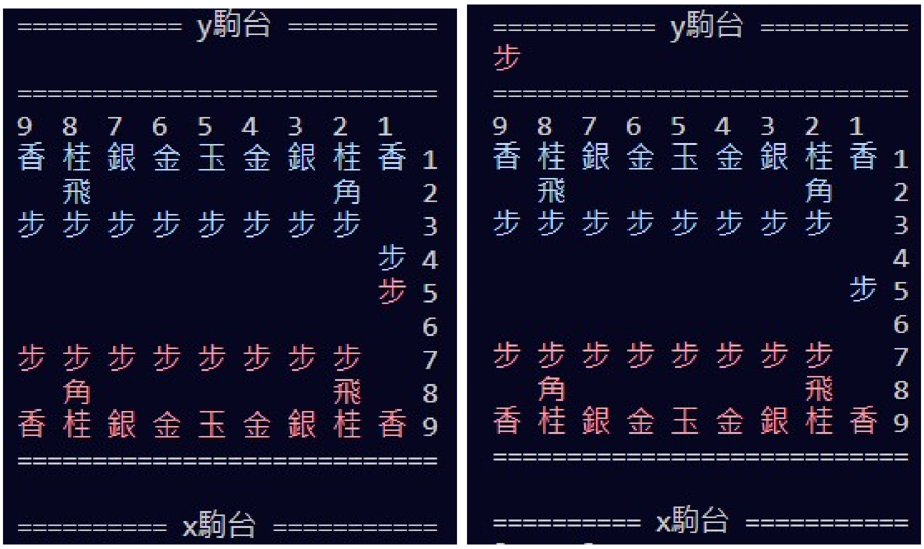


```
[Turn1]
[Player x]
Please enter the initial position:8 8
Select 角 [8][8]

Please enter the new position:6 6
You can not move to here.
[Error 4]:There is chess in your moving path.
Please try again.

Please enter the new position:
```

```
=========== y駒台 ==========

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

========== x駒台 ===========
```

figure.8(a)(b) - (a)Error message(*Bishop* move)(b)example board

Eat goma



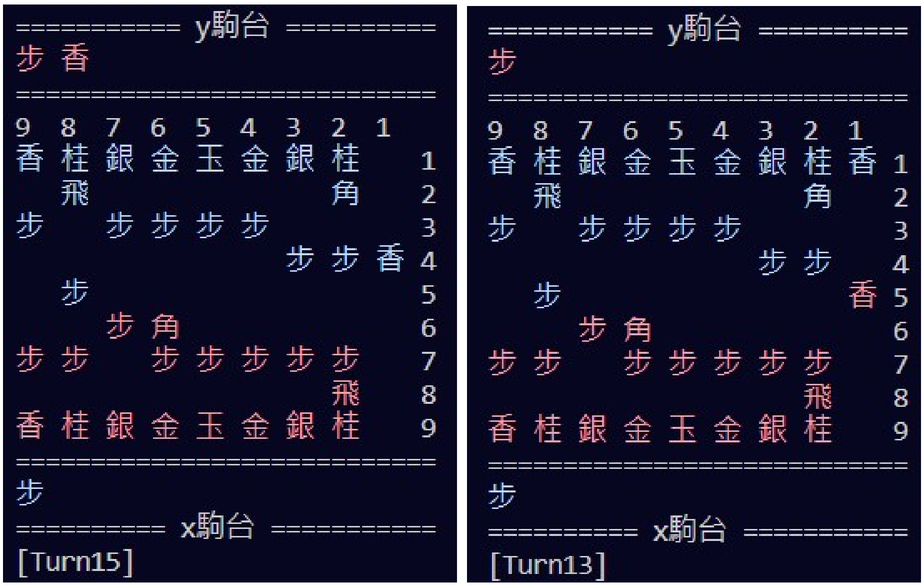
```
=========== y駒台 ==========

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

========== x駒台 ===========
```

```
=========== y駒台 ==========
步
============================
9  8  7  6  5  4  3  2  1
香 桂 銀 金 玉 金 銀 桂 香 1
   飛                角    2
步 步 步 步 步 步 步 步    3
                           4
                        步 5
                           6
步 步 步 步 步 步 步 步    7
   角                飛    8
香 桂 銀 金 玉 金 銀 桂 香 9
============================

========== x駒台 ===========
```

figure.9(a)(b) - (a)previous board(b)after eating chess

1. Cover the chess be ate by the present player's chess.
2. Pick the eaten chess into player's *mochigoma* bag.

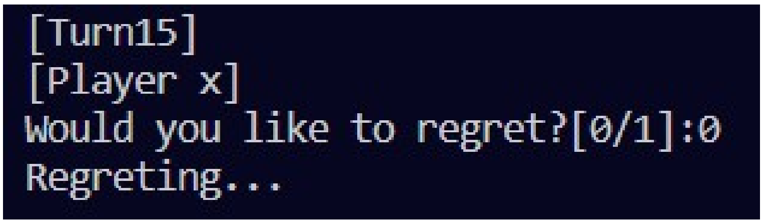
Regret movement



```
=========== y駒台 ==========
步 香
============================
9  8  7  6  5  4  3  2  1
香 桂 銀 金 玉 金 銀 桂    1
   飛                角    2
步    步 步 步 步          3
                  步 步 香 4
   步                      5
      步 角                6
步 步    步 步 步 步 步    7
                     飛    8
香 桂 銀 金 玉 金 銀 桂    9
============================
步
========== x駒台 ===========
[Turn15]
```

```
=========== y駒台 ==========
步
============================
9  8  7  6  5  4  3  2  1
香 桂 銀 金 玉 金 銀 桂 香 1
   飛                角    2
步    步 步 步 步          3
                  步 步    4
   步                   香 5
      步 角                6
步 步    步 步 步 步 步    7
                     飛    8
香 桂 銀 金 玉 金 銀 桂    9
============================
步
========== x駒台 ===========
[Turn13]
```

figure.10(a)(b) - (a)previous board(b)after regretting



```
[Turn15]
[Player x]
Would you like to regret?[0/1]:0
Regreting...
```

figure.11 - Asking for regretting movement

1. Trace back to the previous turn of the player regretting.
2. If any chess had been eaten within the period,
   place it back to the board.
3. Display of turn will minus 2.(e.g. 17 to 15)

3 To-Improve

About file saving...

1. Visualize the game record.
   (a) Make the game record more visualization.
   (b) Figure out how to read csv file.
2. Detect available path when got which chess to move, before enter the goal position.
   (a) Scan all board.
   (b) Check each position by following assigned chess movement rule.
   (c) Return whether there is any path for the assigned chess to go.
3. Drop chess system
   (a) Assign where to drop (position)
   (b) Assign which chess to drop (By number?)
   (c) Check the drop rule.
   (d) Drop the chess but with opposite color.
      i. Change symbol of chess(including enemy's chess).
4. Promotion system
5. Use NoSQL to save game record.

References

[1] Shogi rule

[2] Output with color in C

[3] pgn

[4] libev tutorial

[5] redis