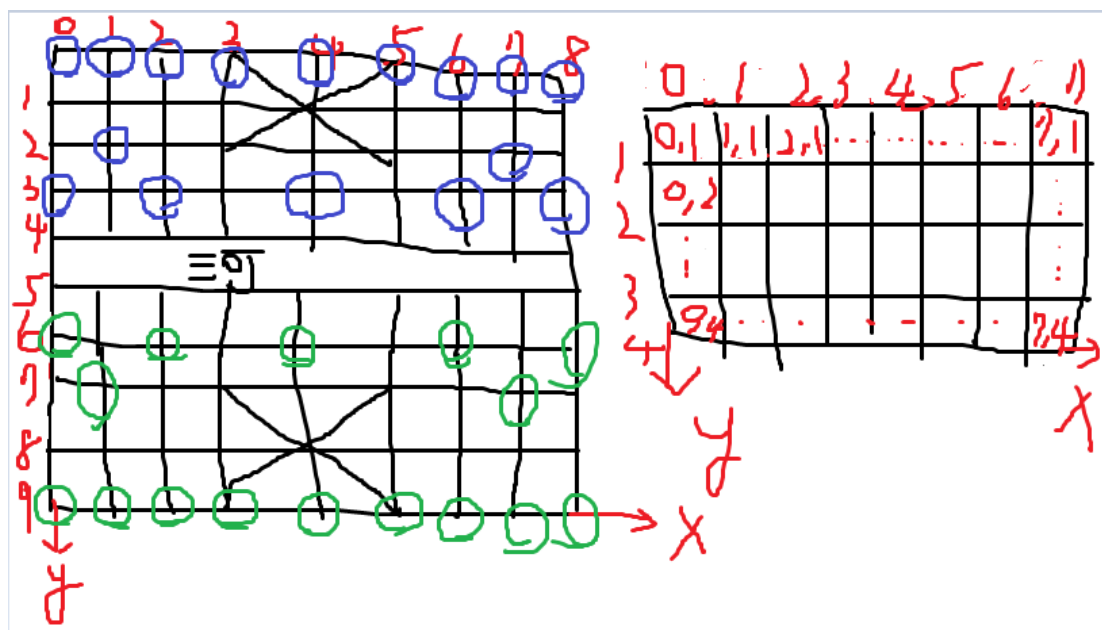


棋盤



| | ChineseChess / DarkChess |
|------|--------------------------|
| 名稱 | o |
| 大小 | o |
| 可吃的子 | o |
| 覆蓋 | x |
| 起始位置 | x |
| 棋盤坐標 | v |

1. 沒寫能不能吃
2. 沒寫能不能移動
3. 最後有一個 test class,分別 new ChineseChess 紅和黑 , new DarkChess 紅和黑 最後印出 起始位置的坐標
4. 其它還沒

Chess class:

1. Variable:

- A. chessName(String) 棋子的名稱(暫定: King, Guard, Bishop, Horse, Rook, Cannon, Pawn)
- B. chessCover(boolean) 是否為覆蓋
- C. boardPosition(position: 含 int x, int y) 在棋盤上的位置
- D. eatRule(String[]) 可以吃的棋子
- E. rank(int) 棋子的大小
- F. startPosition(position) 初始位置
- G. color(String) 紅(r)或黑(b)

2. constructor:

- A. public chess(String name , String color) 存 chessName , color ,呼叫 setRank() 設定其大小 和呼叫 setEatRule() 設定可吃的棋子
--以下暫時注解掉
- B. public chess(String name, String color, position p)
- C. public chess(String name, String color , int x , int y)

3. method:

- A. getChessName: return chessName
- B. setChessName(String):
- C. getChessCover: return Boolean
- D. setChessCover(boolean):
- E. setBoardPosition(position): 暫時沒用到
- F. setBoardPositionX(int x): 暫時沒用到 注解掉
- G. setBoardPositionY(int y): 暫時沒用到 注解掉
- H. getBoardPositionX(): return int
- I. getBoardPositionY(): return int
- J. setStartPosition(position): 暫時沒用到 注解掉

- K. setStartPositionX(int)
- L. setStartPositionY(int)
- M. getStartPositionX(): return int
- N. getStartPositionY(): return int
- O. setColor(String)
- P. setEatRule()
- Q. getEatRule() : return String[] **//問題**
- R. getRank: return int
- S. setRank(): (King=1 , Guard=2, Bishop=3, Rook=4, Horse=5, Cannon=6, Pawn=7)

ChineseChess class:

1. Variable:

- A. Private static int 分為紅色、黑色 棋子的起始位置

2. Constructor:

- A. public chess(String name , String color) 呼叫 super, setStartPosition() 和 setChessCover(false)不覆蓋

--以下暫時注解掉

- B. public chess(String name, String color, position p)
- C. public chess(String name, String color , int x , int y)

3. method:

- A. setStartPosition()

DarkChess class:

3. Variable:

- B. Private static LinkedList<position> 棋子隨機的起始位置

4. Constructor:

- B. public chess(String name , String color) 呼叫 super, setStartPosition() 和 setChessCover(false)不覆蓋

--以下暫時注解掉

- B. public chess(String name, String color, position p)
- C. public chess(String name, String color , int x , int y)

3. method:

- A. setStartPosition()
 - i. Random ran; 隨機亂數
 - ii. Int x , int y 各存亂數產生的 x 和 y
 - iii. Boolean set 不重複並設定
 - iv. position p 存 LinkedList 各 index 得到的 position 型態