He Jin COM S 572 168653509

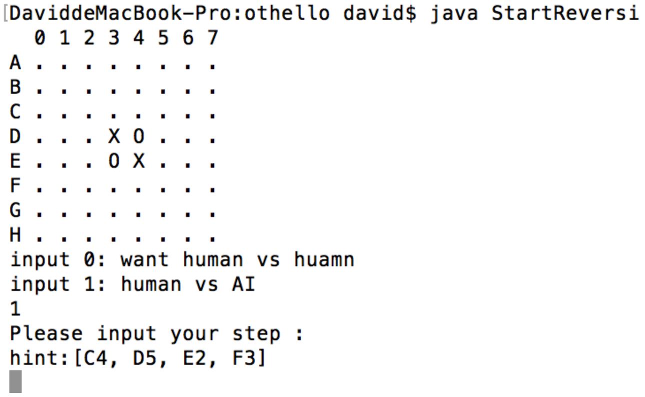
Environment : Mac OS Terminal.

Compile:
javac StartReversi.java

Run:
java StartReversi

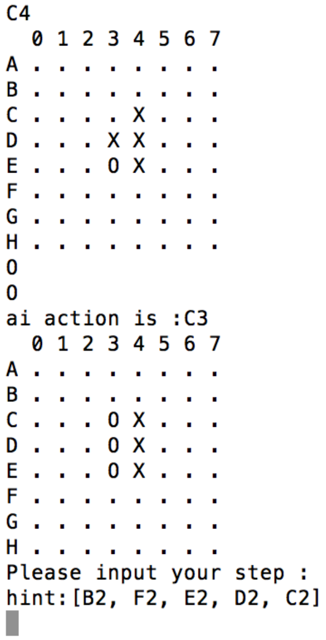
Remind:
The depth of algorithm is set in StartReversi.java
You can search “int depth = 5;” to find it.
BTW, your piece is ‘X’ but not ‘O’.

Screenshot from start … to the end:
1. As you see, there are 2 modes, input 0 for human vs human , input 1 for human vs AI. Here, our lab focuses on AI, then input 1. And every step for yours will be given a hint.



```
[DaviddeMacBook-Pro:othello david$ java StartReversi
   0 1 2 3 4 5 6 7
 A . . . . . . . .
 B . . . . . . . .
 C . . . . . . . .
 D . . . X O . . .
 E . . . O X . . .
 F . . . . . . . .
 G . . . . . . . .
 H . . . . . . . .
 input 0: want human vs huamn
 input 1: human vs AI
 1
 Please input your step :
 hint:[C4, D5, E2, F3]
```

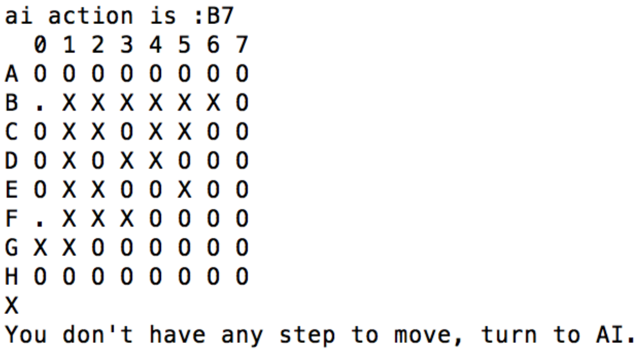
2. If you choose one of steps, then AI will automatically make his own decision.



```
C4
  0 1 2 3 4 5 6 7
A . . . . . . . .
B . . . . . . . .
C . . . . X . . .
D . . . X X . . .
E . . . O X . . .
F . . . . . . . .
G . . . . . . . .
H . . . . . . . .
O
O
ai action is :C3
  0 1 2 3 4 5 6 7
A . . . . . . . .
B . . . . . . . .
C . . . O X . . .
D . . . O X . . .
E . . . O X . . .
F . . . . . . . .
G . . . . . . . .
H . . . . . . . .
Please input your step :
hint:[B2, F2, E2, D2, C2]
```

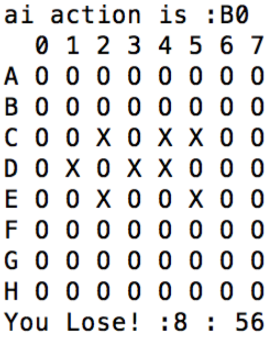
3. BTW, if you choose some invalid step, then system will let you re-input the action.

4. In the pic below, you can see if AI represented by ’O’ makes a decision, and you don’t have any steps, then it will turn to AI’s step.



```
ai action is :B7
  0 1 2 3 4 5 6 7
A O O O O O O O O
B . X X X X X X O
C O X X O X X O O
D O X O X X O O O
E O X X O O X O O
F . X X X O O O O
G X X O O O O O O
H O O O O O O O O
X
You don't have any step to move, turn to AI.
```

5. In the end, if two players do not have any steps either, then game ends & show you results.



```
ai action is :B0
  0 1 2 3 4 5 6 7
A O O O O O O O O
B O O O O O O O O
C O O X O X X O O
D O X O X X O O O
E O O X O O X O O
F O O O O O O O O
G O O O O O O O O
H O O O O O O O O
You Lose! :8 : 56
```

Q5.
As the search depth increases, the wiser the AI make his decision because it can calculate the maximum utility after the depth number of steps based on the evaluation.

(Here I just don’t give the pics, but with the description of it)
I can see it from comparison of the same board and same actions I made at that time, and AI responds with different optimal action at the different depth K. As the depth is assigned to 5 or above, it’s hard to beat AI because it has the vision of what is going to happen in 5 or more steps.

And Evaluation function is equally important. Through my observation, The corner compared to other position is the most important in the early stage, and edge position is the second important position, and then the position in the middle that is the least important. Thus, I designed a board_value[][](in the beginning of Board.java), a two-dimension double array which is assigned values where the corner get highest value, then the edge and middle positions get the lowest value. Although it’s not that scientific, it does make some sense for pieces layout. Through modify some positions’ value, it can improve or influence the Alphabeta algorithm decision. What I did is to find the proper value for these positions which can let it do more reasonable action.