# E03 Othello Game (α – β pruning)

| 姓名 | 学号 | 日期 |
| --- | --- | --- |
| 何泽 | 18340052 | 2020.09.20 |

## 目录



## 1 Othello

Othello (or Reversi) is a strategy board game for two players, played on an 8 × 8 uncheckered board. There are sixty-four identical game pieces called disks (often spelled ″discs″), which are light on one side and dark on the other. Please see figure 1.

Players take turns placing disks on the board with their assigned color facing up. During a play, any disks of the opponent's color that are in a straight line and bounded by the disk just placed and another disk of the current player's color are turned over to the current player's color.

The object of the game is to have the majority of disks turned to display your color when the last playable empty square is filled.

You can refer to http://www.tothello.com/html/guideline_of_reversed_othello.html for more information of guideline, meanwhile, you can download the software to have a try from http://www.tothello.com/html/download.html. The game installer tothello trial setup.exe can also be found in the current folder.

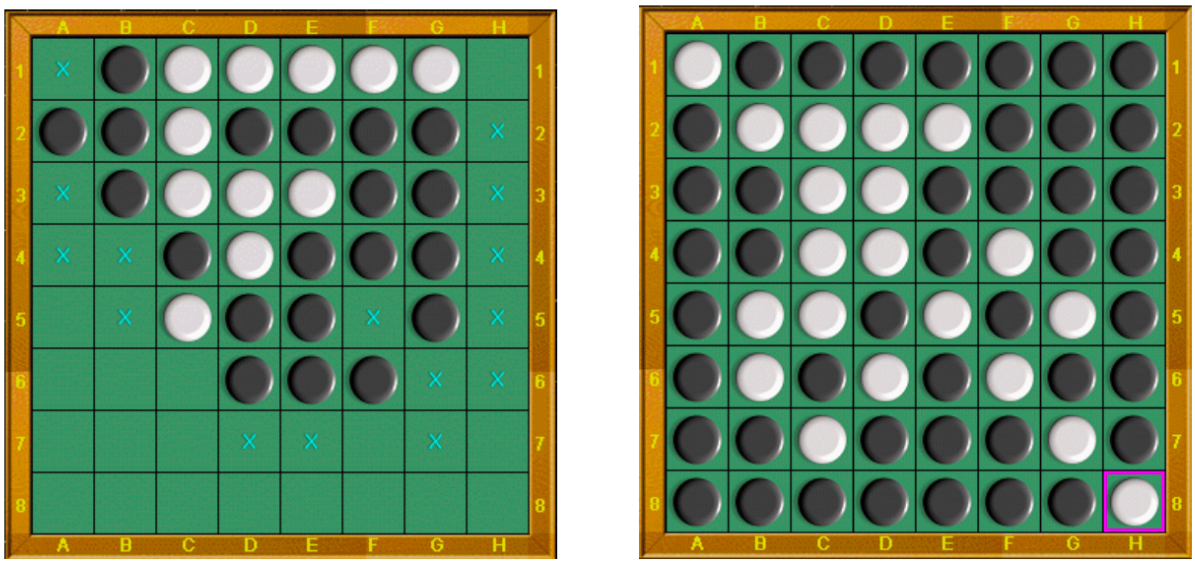


Figure 1: Othello Game

## 2 Tasks

1. In order to reduce the complexity of the game, we think the board is 6 × 6.
2. There are several evaluation functions that involve many aspects, you can turn to http://www.cs.cornell.edu/~yuli/othello/othello.html for help. In order to reduce the difficulty of the task, I have gaven you some hints of evaluation function in the file Heuristic Function for Reversi (Othello).cpp.
3. Please choose an appropriate evaluation function and use min-max and α – β prunning to implement the Othello game. The framework file you can refer to is Othello.cpp. Of course, I wish your program can beat the computer.
4. Write the related codes and take a screenshot of the running results in the file named E03_StudentNumber.pdf, and send it to ai2020@foxmail.com, the deadline is 2020.09.20 23:59:59.

## 3 Codes

- 我是保留了原来的评价函数，然后在提供的新的评价函数基础上重写了一个新的评价函数，然后让这两种评价函数进行PK
- 对于min-max 和 α – β 剪枝算法，我使用的就是提供的代码

```
if (num == 0) {    /* 无处落子 */
        if (board->Rule(board, (enum Option) - player)){
 /* 对方可以落子,让对方下.*/
            Othello tempBoard;
            Do nextChoice;
            Do *pNextChoice = &nextChoice;
            board->Copy(&tempBoard, board);
            pNextChoice = Find(&tempBoard, (enum Option) -
player, step - 1, -max, -min, pNextChoice,who_judge);
            choice->score = -pNextChoice->score;
            choice->pos.first = -1;
            choice->pos.second = -1;
            return choice;
        }
        else{     /* 对方也无处落子,游戏结束. */
            int value = WHITE*(board->whiteNum) + BLACK*
(board->blackNum);
            if (player*value>0)
                choice->score = MAX - 1;
            else if (player*value<0)
                choice->score = -MAX + 1;
            else
                choice->score = 0;
            return choice;
        }
    }

    if (step ≤ 0){
    if (who_judge)
      choice->score = board->MyJudge(board, player);
    else
      choice->score = board->Judge(board, player);
    return choice;
  }

    allChoices = (Do *)malloc(sizeof(Do)*num);
    k = 0;
    for (i = 0; i<6; i++){
```

```
36          for (j = 0; j<6; j++){
37              if (i == 0 || i == 5 || j == 0 || j == 5){
38                  if (board->cell[i][j].color == SPACE &&
    board->cell[i][j].stable){
39                      allChoices[k].score = -MAX;
40                      allChoices[k].pos.first = i;
41                      allChoices[k].pos.second = j;
42                      k++;
43                  }
44              }
45          }
46      }
47      for (i = 0; i<6; i++){
48          for (j = 0; j<6; j++){
49              if ((i == 2 || i == 3 || j == 2 || j == 3) && (i
    ≥ 2 && i ≤ 3 && j ≥ 2 && j ≤ 3)){
50                  if (board->cell[i][j].color == SPACE &&
    board->cell[i][j].stable){
51                      allChoices[k].score = -MAX;
52                      allChoices[k].pos.first = i;
53                      allChoices[k].pos.second = j;
54                      k++;
55                  }
56              }
57          }
58      }
59      for (i = 0; i<6; i++){
60          for (j = 0; j<6; j++){
61              if ((i == 1 || i == 4 || j == 1 || j == 4) && (i
    ≥ 1 && i ≤ 4 && j ≥ 1 && j ≤ 4)){
62                  if (board->cell[i][j].color == SPACE &&
    board->cell[i][j].stable){
63                      allChoices[k].score = -MAX;
64                      allChoices[k].pos.first = i;
65                      allChoices[k].pos.second = j;
66                      k++;
67                  }
68              }
69          }
```

```
    }
    for (k = 0; k<num; k++){
        Othello tempBoard;
        Do thisChoice, nextChoice;
        Do *pNextChoice = &nextChoice;
        thisChoice = allChoices[k];
        board->Copy(&tempBoard, board);
        board->Action(&tempBoard, &thisChoice, player);
        pNextChoice = Find(&tempBoard, (enum Option) -
player, step - 1, -max, -min, pNextChoice,who_judge);
        thisChoice.score = -pNextChoice->score;

        if (thisChoice.score>min && thisChoice.score<max){
 /* 可以预计的更优值 */
            min = thisChoice.score;
            choice->score = thisChoice.score;
            choice->pos.first = thisChoice.pos.first;
            choice->pos.second = thisChoice.pos.second;
        }
        else if (thisChoice.score ≥ max) {   /* 好的超乎预计
*/
            choice->score = thisChoice.score;
            choice->pos.first = thisChoice.pos.first;
            choice->pos.second = thisChoice.pos.second;
            break;
        }
        /* 不如已知最优值 */
    }
```

- 而评价函数我是在提供的基础上改的，将8x8改为了6x6，并去掉了最后的l

```
double Othello::MyJudge(Othello *board, enum Option player)
{
  int my_tiles = 0, opp_tiles = 0, i, j, k, my_front_tiles =
0, opp_front_tiles = 0, x, y;
  double p = 0, c = 0, l = 0, m = 0, f = 0, d = 0;
  enum Option my_color = player;
  enum Option opp_color = (enum Option) - player;

```

```
  // eight directions
  int X1[] = {-1, -1, 0, 1, 1, 1, 0, -1};
  int Y1[] = {0, 1, 1, 1, 0, -1, -1, -1};
  // aprior weights for each movement
  int V[6][6]= {{20, -3, 11, 11, -3, 20},{-3, -7, -4, -4, -7, -3},{11, -4, 2, 2, -4, 11},{11, -4, 2, 2, -4, 11},{-3, -7, -4, -4, -7, -3},{20, -3, 11, 11, -3, 20}};

  // Piece difference
  for (i = 0; i < 6; i++)
    for (j = 0; j < 6; j++){
      // count how many tiles are occupied
      if (board->cell[i][j].color == my_color){
        d += V[i][j];
        my_tiles++;
      }
      else if (board->cell[i][j].color == opp_color){
        d -= V[i][j];
        opp_tiles++;
      }
      // find the difference in eight directions
      if (board->cell[i][j].color != SPACE){
        for (k = 0; k < 8; k++){
          x = i + X1[k];
          y = j + Y1[k];
          if (x >= 0 && x < 6 && y >= 0 && y < 6 && board->cell[x][y].color == SPACE){
            if (board->cell[i][j].color == my_color)
              my_front_tiles++;
            else
              opp_front_tiles++;
            break;
          }
        }
      }
    }
  // calculate the proportions
  if (my_tiles > opp_tiles)
    p = (100.0 * my_tiles) / (my_tiles + opp_tiles);
```

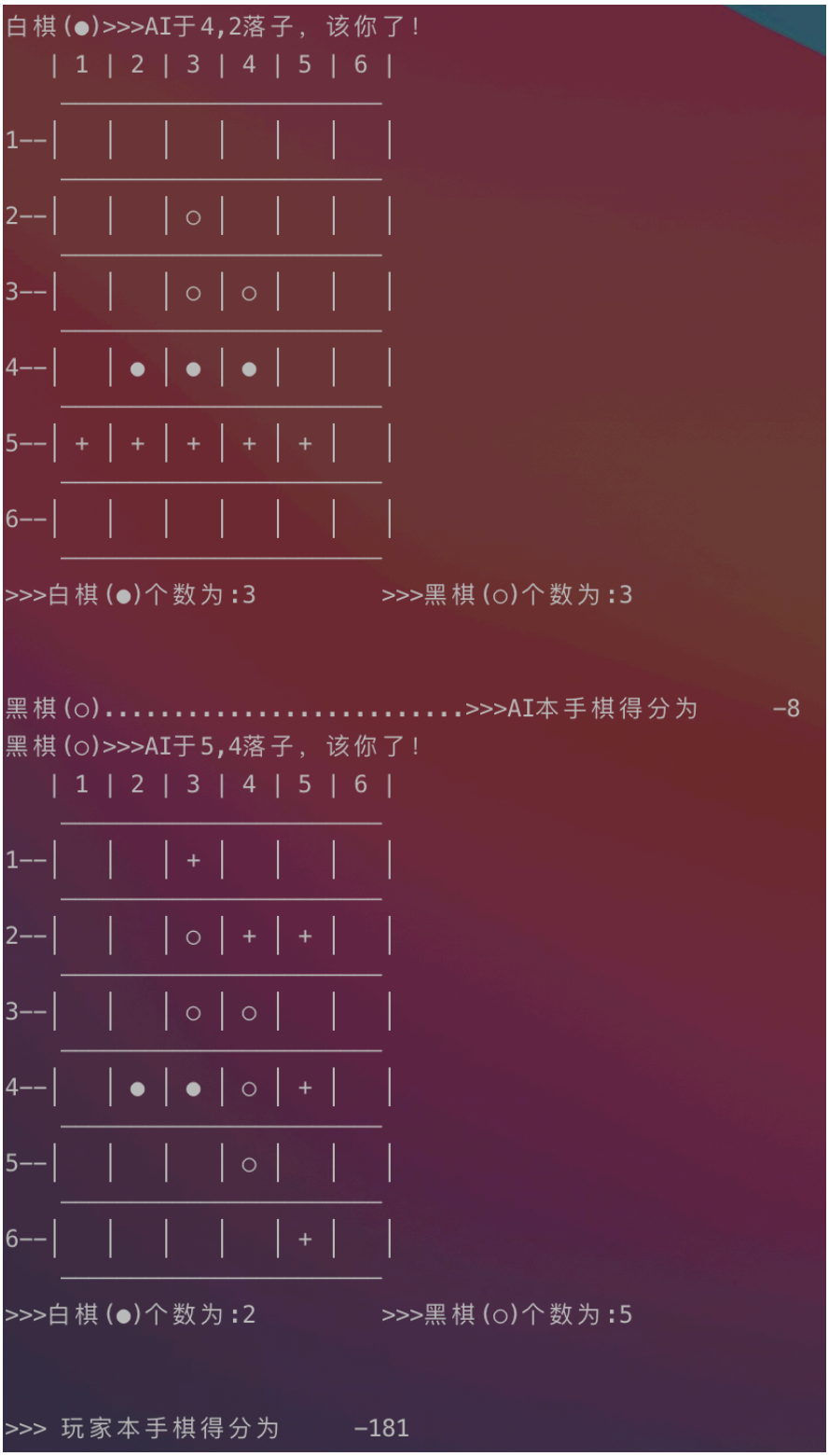
```
  else if (my_tiles < opp_tiles)
    p = -(100.0 * opp_tiles) / (my_tiles + opp_tiles);
  else
    p = 0;

  if (my_front_tiles > opp_front_tiles)
    f = -(100.0 * my_front_tiles) / (my_front_tiles + opp_front_tiles);
  else if (my_front_tiles < opp_front_tiles)
    f = (100.0 * opp_front_tiles) / (my_front_tiles + opp_front_tiles);
  else
    f = 0;

  // Corner occupancy
  my_tiles = opp_tiles = 0;
  if (board->cell[0][0].color == my_color)
    my_tiles++;
  else if (board->cell[0][0].color == opp_color)
    opp_tiles++;
  if (board->cell[0][5].color == my_color)
    my_tiles++;
  else if (board->cell[0][5].color == opp_color)
    opp_tiles++;
  if (board->cell[5][0].color == my_color)
    my_tiles++;
  else if (board->cell[5][0].color == opp_color)
    opp_tiles++;
  if (board->cell[5][5].color == my_color)
    my_tiles++;
  else if (board->cell[5][5].color == opp_color)
    opp_tiles++;
  c = 25 * (my_tiles - opp_tiles);

  // Mobility
  // The more tiles can be moved on, the better
  my_tiles = Rule(board, my_color);
  opp_tiles = Rule(board, opp_color);
  if (my_tiles > opp_tiles)
```

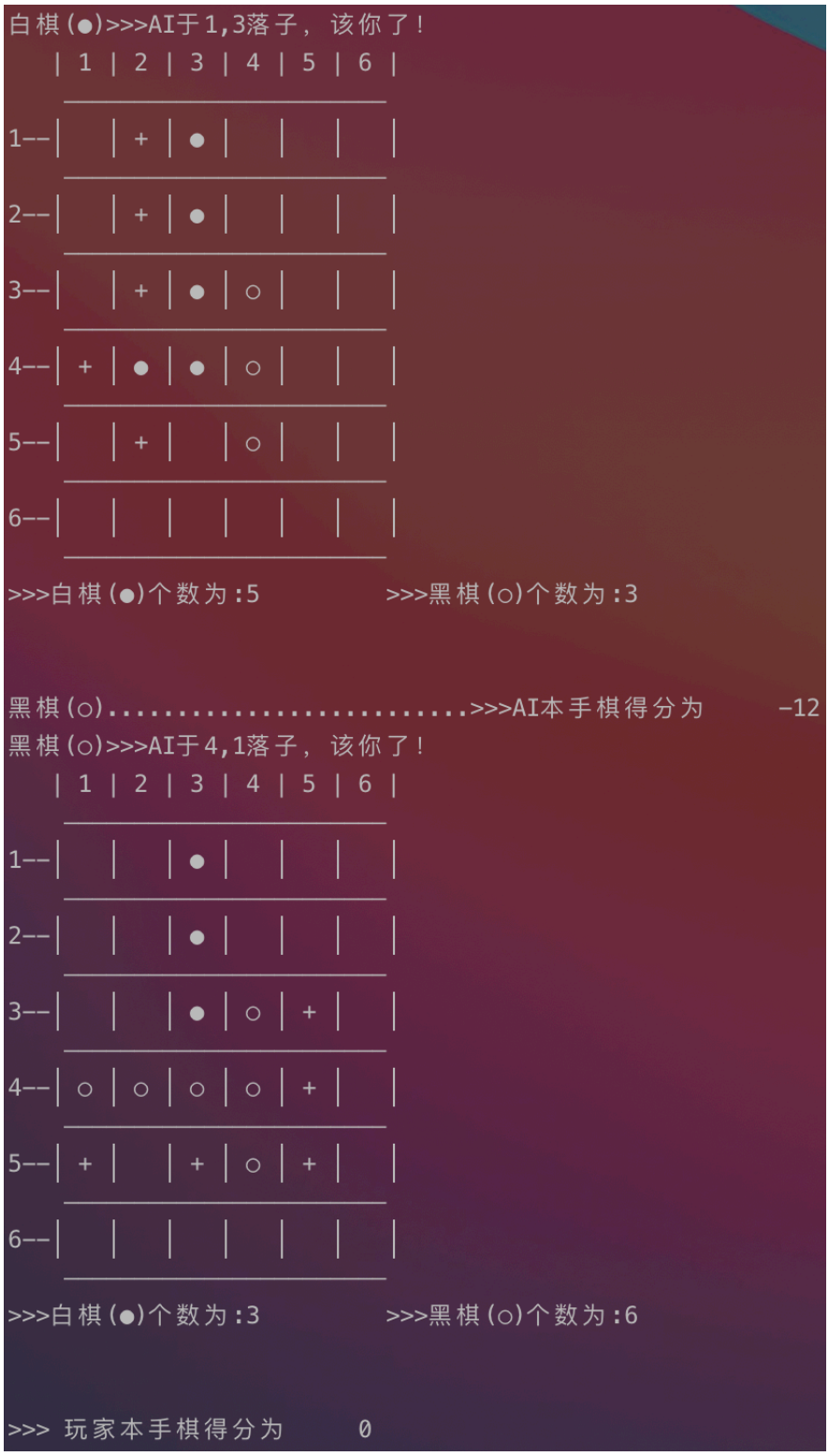
```
81      m = (100.0 * my_tiles) / (my_tiles + opp_tiles);
82    else if (my_tiles < opp_tiles)
83      m = -(100.0 * opp_tiles) / (my_tiles + opp_tiles);
84    else
85      m = 0;
86
87    // final weighted score (magic numbers!)
88    double score = (10 * p) + (801.724 * c) +  (78.922 * m) +
  (74.396 * f) + (10 * d);
89    return score;
90  }
```

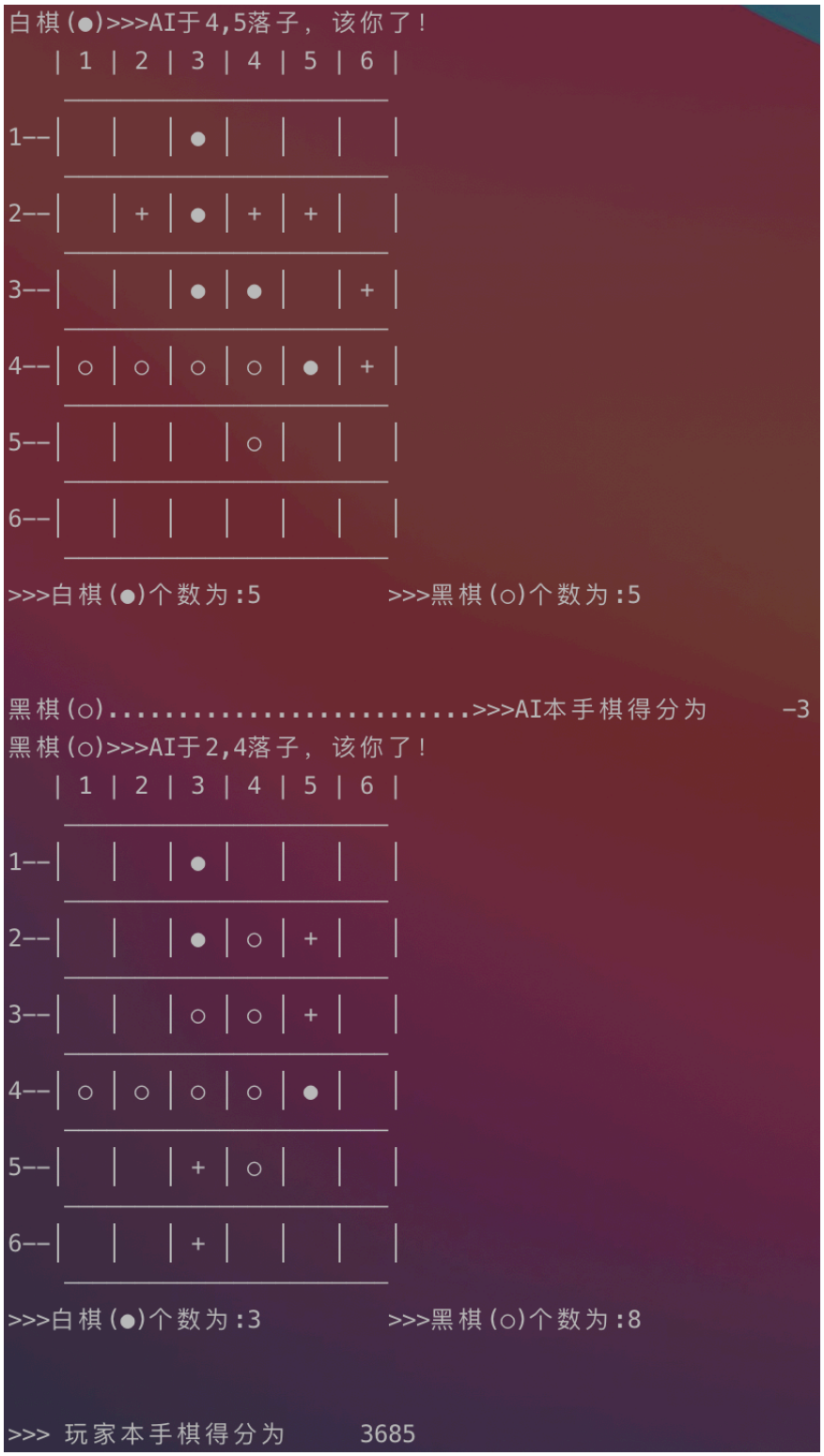
## 4 Results

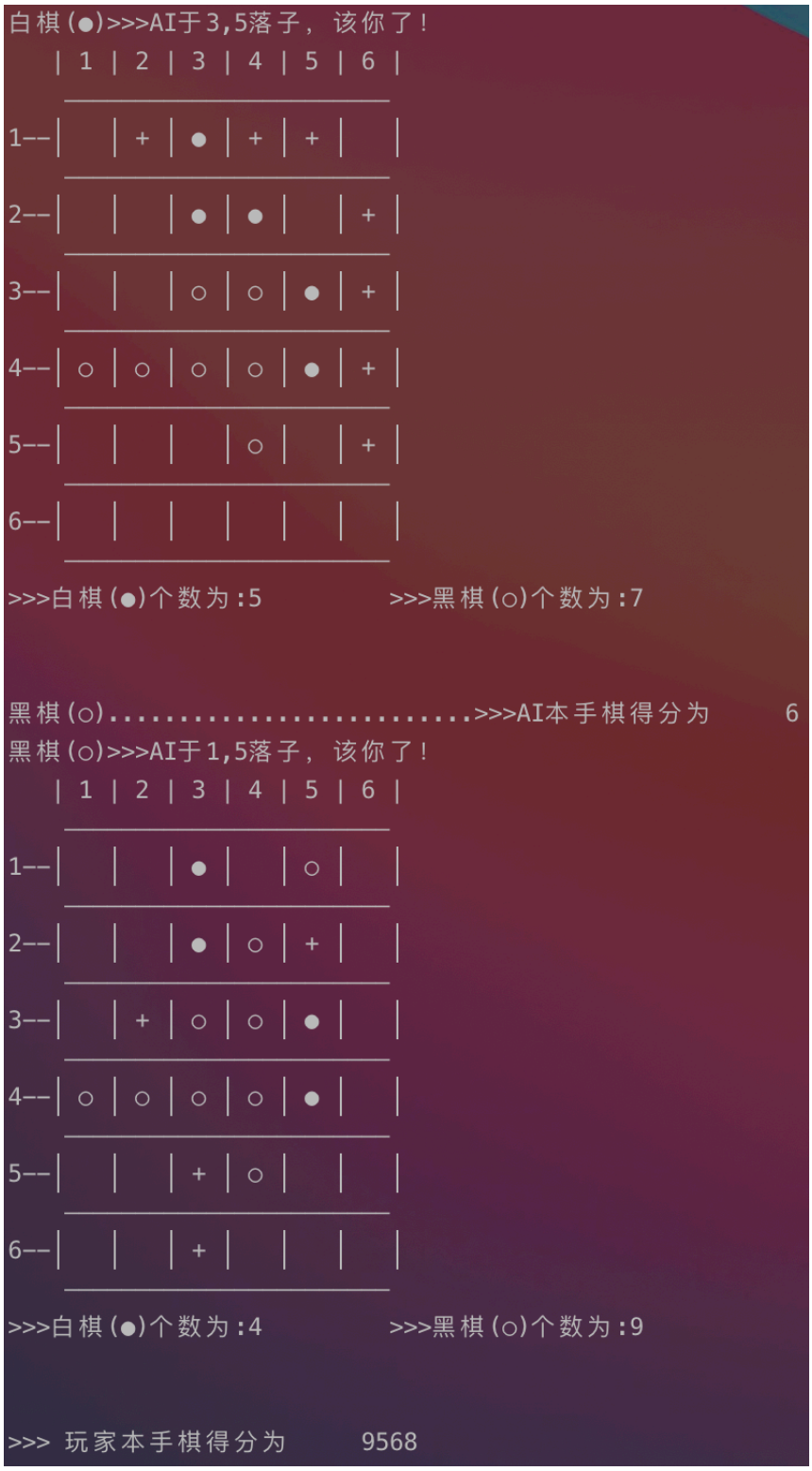
使用两种算法进行PK，可以看到在新的评价函数下的算法打败了原来的，下面记录了每次的过程，我为我的算法选了白棋，原来的是黑棋

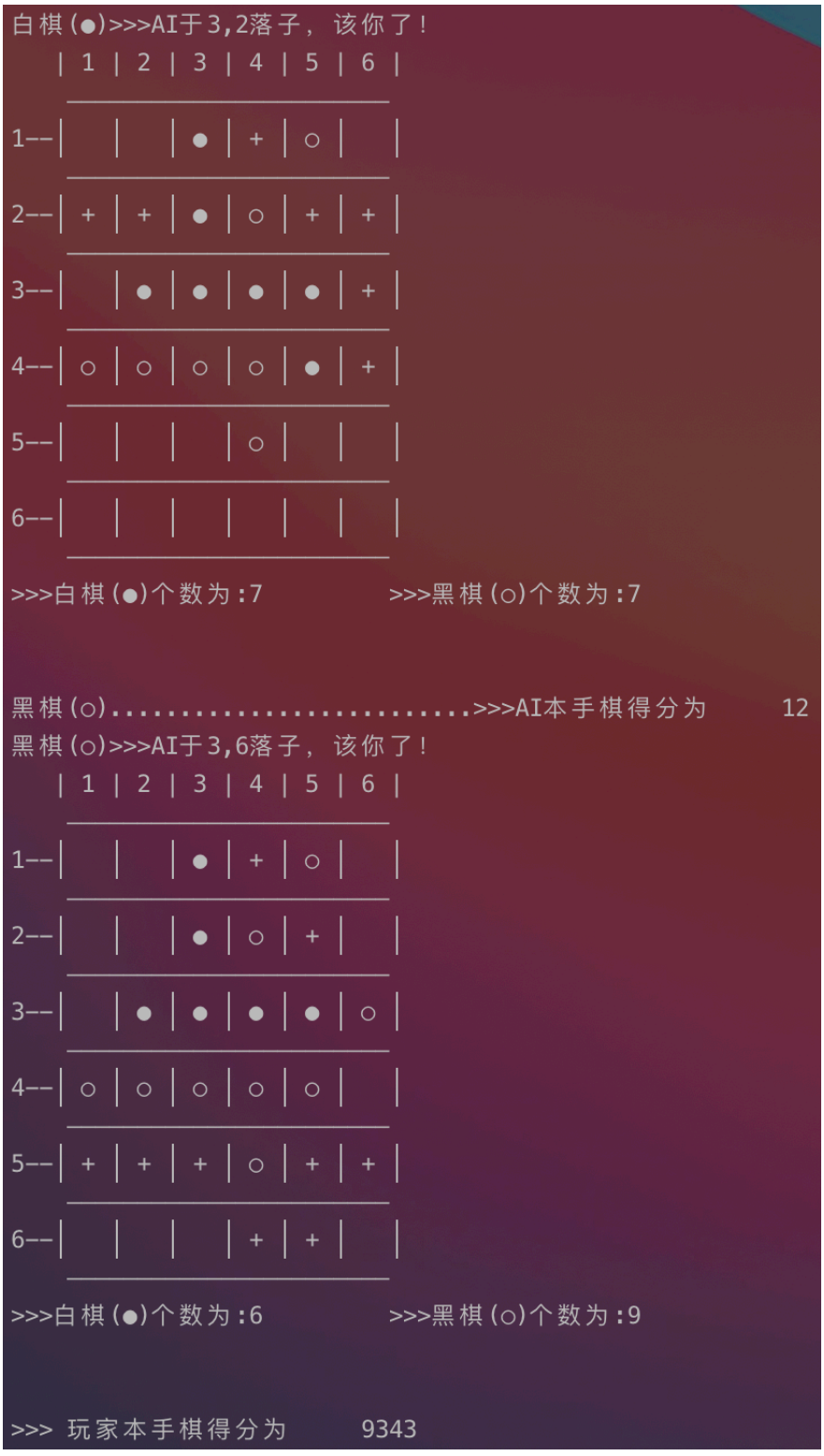
```
>>>人机对战开始:
>>>请选择执黑棋(○),或执白棋(●): 输入1为黑棋, -1为白棋: -1
>>>黑棋行动:

   | 1 | 2 | 3 | 4 | 5 | 6 |
   ┌───┬───┬───┬───┬───┬───┐
1--|   |   |   |   |   |   |
   ├───┼───┼───┼───┼───┼───┤
2--|   |   | + |   |   |   |
   ├───┼───┼───┼───┼───┼───┤
3--|   | + | ● | ○ |   |   |
   ├───┼───┼───┼───┼───┼───┤
4--|   |   | ○ | ● | + |   |
   ├───┼───┼───┼───┼───┼───┤
5--|   |   |   | + |   |   |
   ├───┼───┼───┼───┼───┼───┤
6--|   |   |   |   |   |   |
   └───┴───┴───┴───┴───┴───┘
>>>白棋(●)个数为:2          >>>黑棋(○)个数为:2


黑棋(○).........................>>>AI本手棋得分为      -7
黑棋(○)>>>AI于2,3落子, 该你了!
   | 1 | 2 | 3 | 4 | 5 | 6 |
   ┌───┬───┬───┬───┬───┬───┐
1--|   |   |   |   |   |   |
   ├───┼───┼───┼───┼───┼───┤
2--|   | + | ○ | + |   |   |
   ├───┼───┼───┼───┼───┼───┤
3--|   |   | ○ | ○ |   |   |
   ├───┼───┼───┼───┼───┼───┤
4--|   | + | ○ | ● |   |   |
   ├───┼───┼───┼───┼───┼───┤
5--|   |   |   |   |   |   |
   ├───┼───┼───┼───┼───┼───┤
6--|   |   |   |   |   |   |
   └───┴───┴───┴───┴───┴───┘
>>>白棋(●)个数为:1          >>>黑棋(○)个数为:4


>>> 玩家本手棋得分为      -210
```

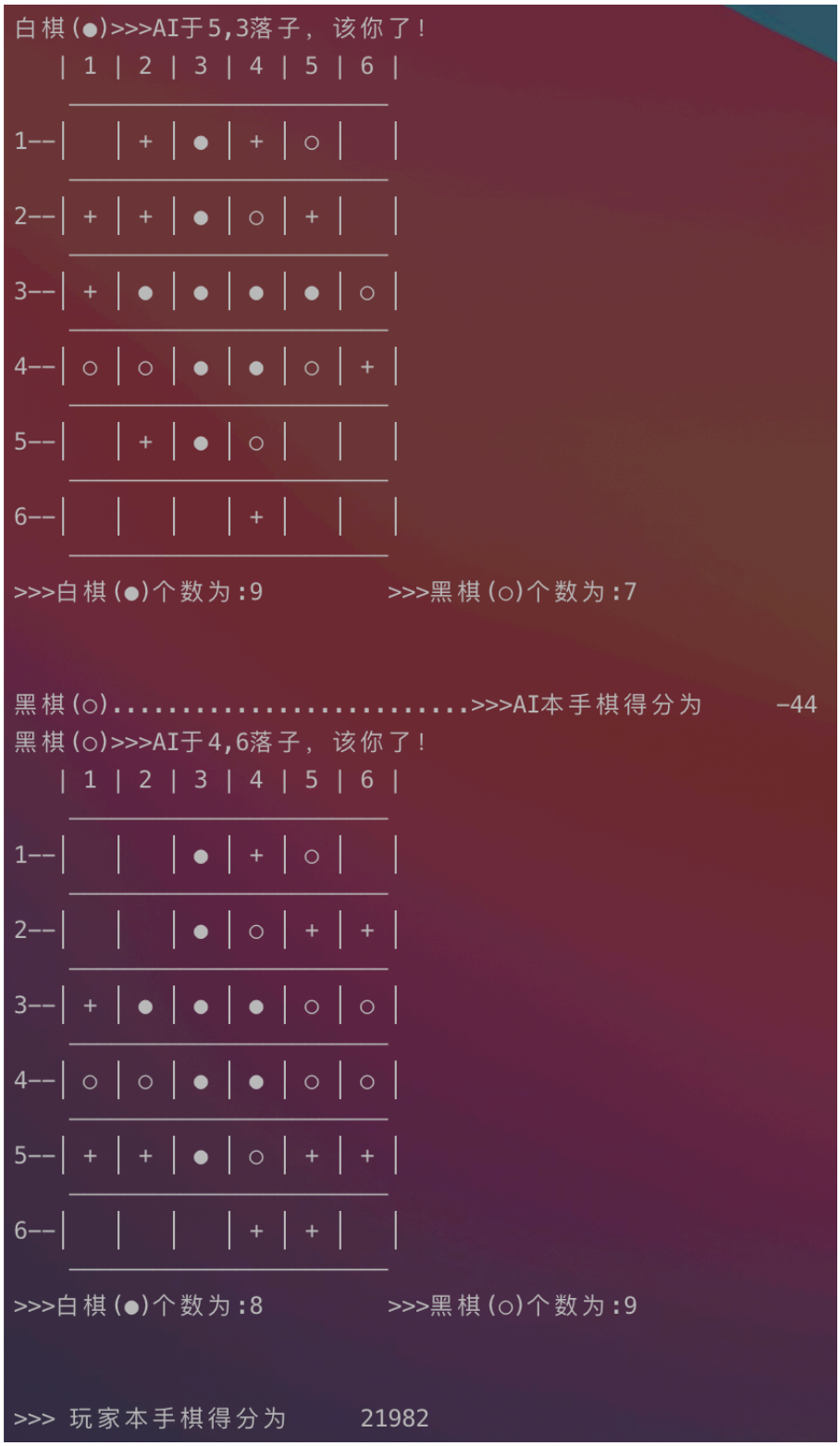

白棋(●)>>>AI于4,2落子，该你了！
| 1 | 2 | 3 | 4 | 5 | 6 |
1--| | | | | | |
2--| | | ○ | | | |
3--| | | ○ | ○ | | |
4--| | ● | ● | ● | | |
5--| + | + | + | + | + | |
6--| | | | | | |
>>>白棋(●)个数为:3
>>>黑棋(○)个数为:3
黑棋(○).....>>>AI本手棋得分为 -8
黑棋(○)>>>AI于5,4落子，该你了！
| 1 | 2 | 3 | 4 | 5 | 6 |
1--| | | + | | | |
2--| | | ○ | + | + | |
3--| | | ○ | ○ | | |
4--| | ● | ● | ○ | + | |
5--| | | | ○ | | |
6--| | | | | + | |
>>>白棋(●)个数为:2
>>>黑棋(○)个数为:5
>>> 玩家本手棋得分为 -181

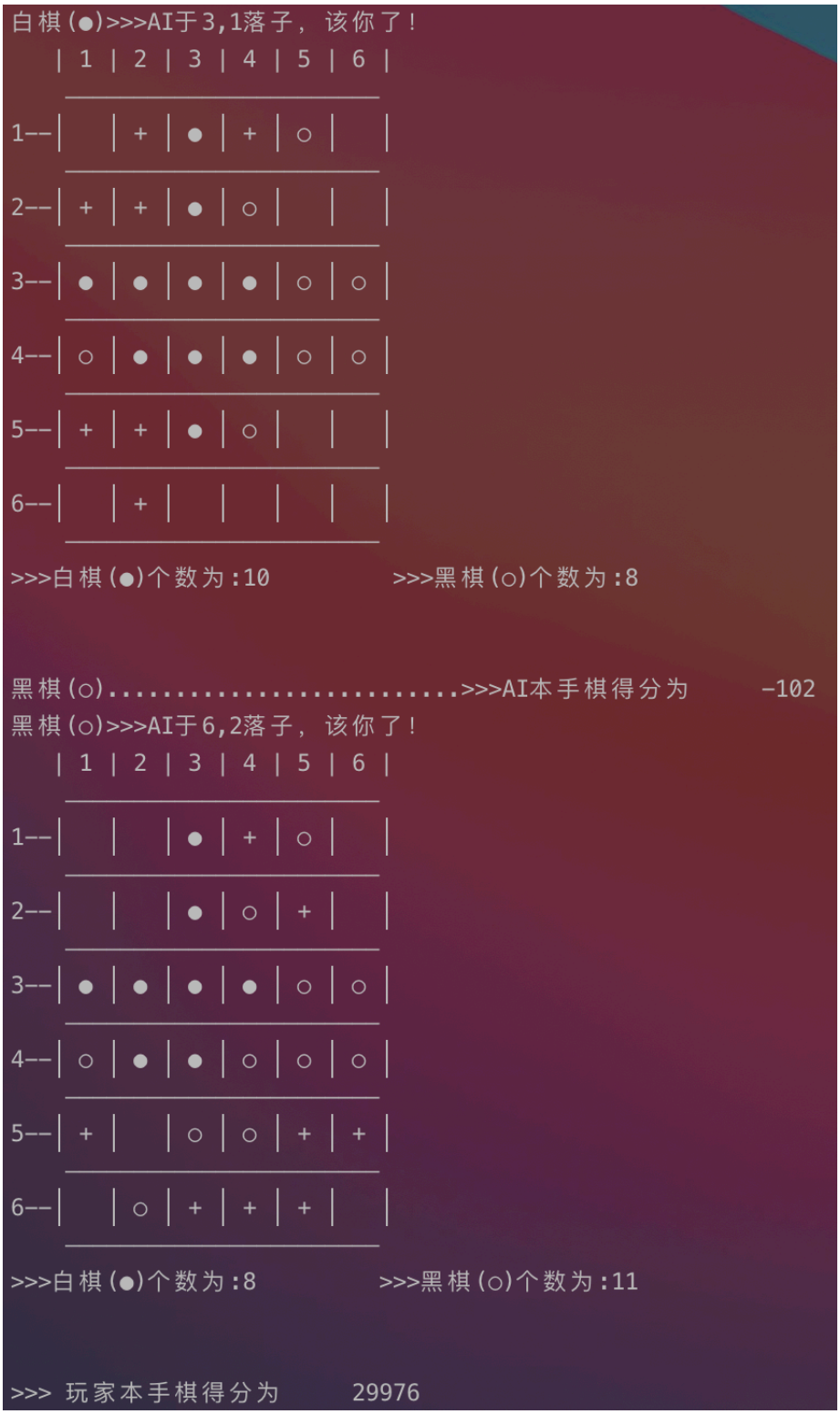

白棋(●)>>>AI于1,3落子，该你了！
| 1 | 2 | 3 | 4 | 5 | 6 |
1--| | + | ● | | | |
2--| | + | ● | | | |
3--| | + | ● | ○ | | |
4--| + | ● | ● | ○ | | |
5--| | + | | ○ | | |
6--| | | | | | |
>>>白棋(●)个数为:5
>>>黑棋(○)个数为:3
黑棋(○).....>>>AI本手棋得分为 -12
黑棋(○)>>>AI于4,1落子，该你了！
| 1 | 2 | 3 | 4 | 5 | 6 |
1--| | | ● | | | |
2--| | | ● | | | |
3--| | | ● | ○ | + | |
4--| ○ | ○ | ○ | ○ | + | |
5--| + | | + | ○ | + | |
6--| | | | | | |
>>>白棋(●)个数为:3
>>>黑棋(○)个数为:6
>>> 玩家本手棋得分为 0


白棋(●)>>>AI于4,5落子，该你了！
| 1 | 2 | 3 | 4 | 5 | 6 |
1--| | | ● | | | |
2--| | + | ● | + | + | |
3--| | | ● | ● | | + |
4--| ○ | ○ | ○ | ○ | ● | + |
5--| | | | ○ | | |
6--| | | | | | |
>>>白棋(●)个数为:5
>>>黑棋(○)个数为:5
黑棋(○).....>>>AI本手棋得分为 -3
黑棋(○)>>>AI于2,4落子，该你了！
| 1 | 2 | 3 | 4 | 5 | 6 |
1--| | | ● | | | |
2--| | | ● | ○ | + | |
3--| | | ○ | ○ | + | |
4--| ○ | ○ | ○ | ○ | ● | |
5--| | | + | ○ | | |
6--| | | + | | | |
>>>白棋(●)个数为:3
>>>黑棋(○)个数为:8
>>> 玩家本手棋得分为 3685

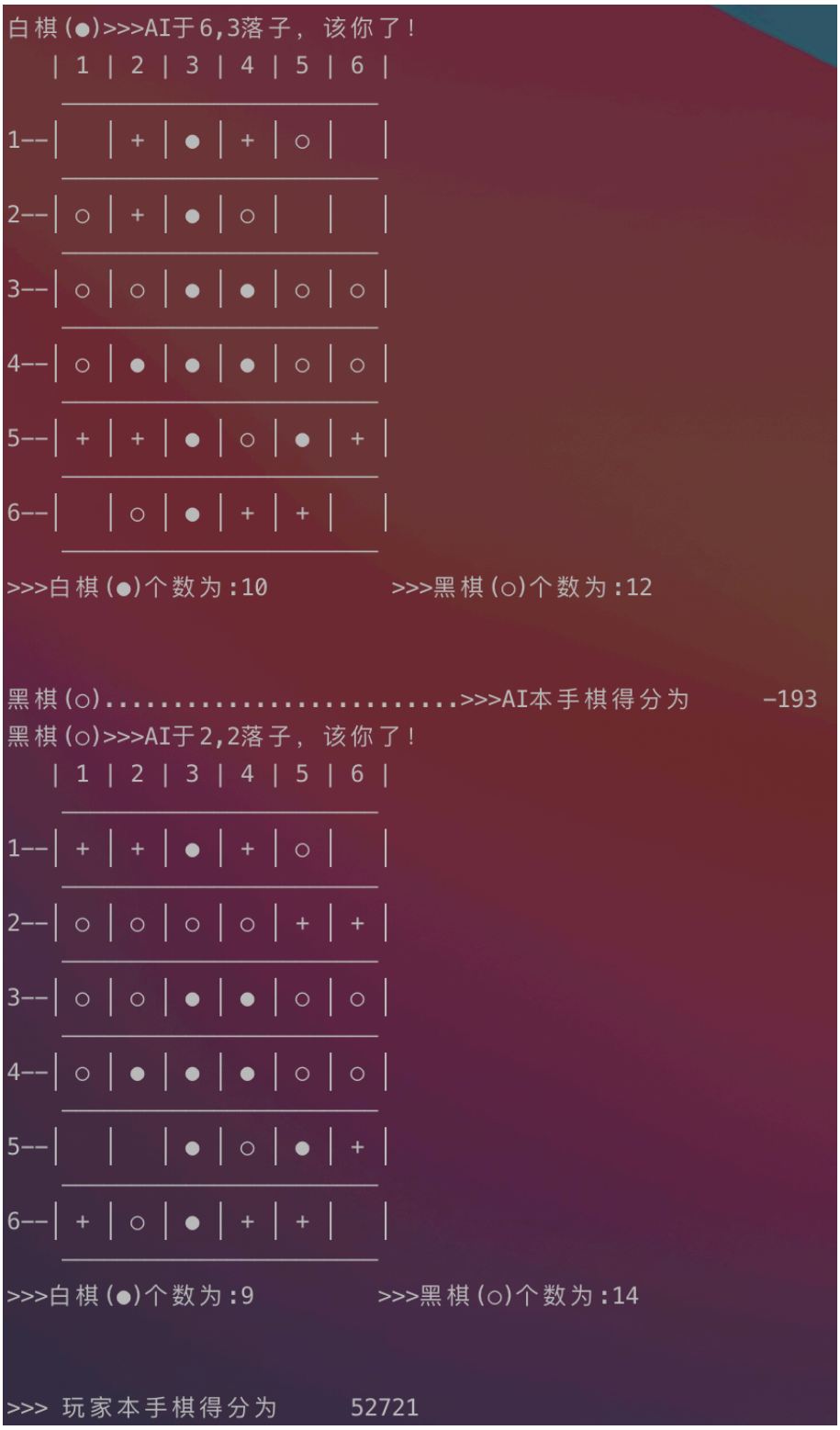

白棋(●)>>>AI于3,5落子，该你了！
| 1 | 2 | 3 | 4 | 5 | 6 |
1--| | + | ● | + | + | |
2--| | | ● | ● | | + |
3--| | | ○ | ○ | ● | + |
4--| ○ | ○ | ○ | ○ | ● | + |
5--| | | | ○ | | + |
6--| | | | | | |
>>>白棋(●)个数为:5
>>>黑棋(○)个数为:7
黑棋(○).....>>>AI本手棋得分为 6
黑棋(○)>>>AI于1,5落子，该你了！
| 1 | 2 | 3 | 4 | 5 | 6 |
1--| | | ● | | ○ | |
2--| | | ● | ○ | + | |
3--| | + | ○ | ○ | ● | |
4--| ○ | ○ | ○ | ○ | ● | |
5--| | | + | ○ | | |
6--| | | + | | | |
>>>白棋(●)个数为:4
>>>黑棋(○)个数为:9
>>> 玩家本手棋得分为 9568

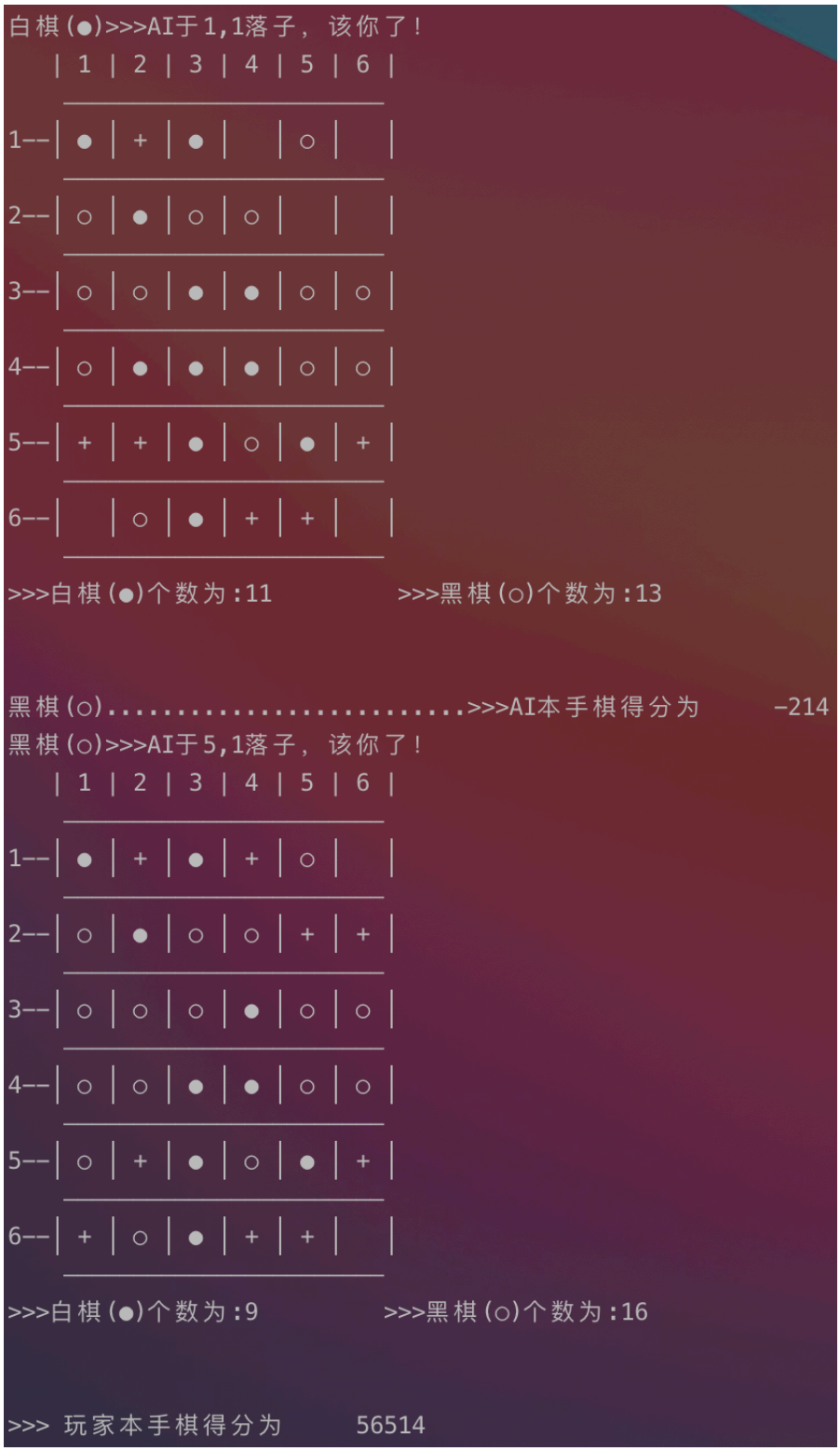

白棋(●)>>>AI于3,2落子，该你了！
| 1 | 2 | 3 | 4 | 5 | 6 |
1--| | | ● | + | ○ | |
2--| + | + | ● | ○ | + | + |
3--| | ● | ● | ● | ● | + |
4--| ○ | ○ | ○ | ○ | ● | + |
5--| | | | ○ | | |
6--| | | | | | |
>>>白棋(●)个数为:7
>>>黑棋(○)个数为:7
黑棋(○).....>>>AI本手棋得分为 12
黑棋(○)>>>AI于3,6落子，该你了！
| 1 | 2 | 3 | 4 | 5 | 6 |
1--| | | ● | + | ○ | |
2--| | | ● | ○ | + | |
3--| | ● | ● | ● | ● | ○ |
4--| ○ | ○ | ○ | ○ | ○ | |
5--| + | + | + | ○ | + | + |
6--| | | | + | + | |
>>>白棋(●)个数为:6
>>>黑棋(○)个数为:9
>>> 玩家本手棋得分为 9343

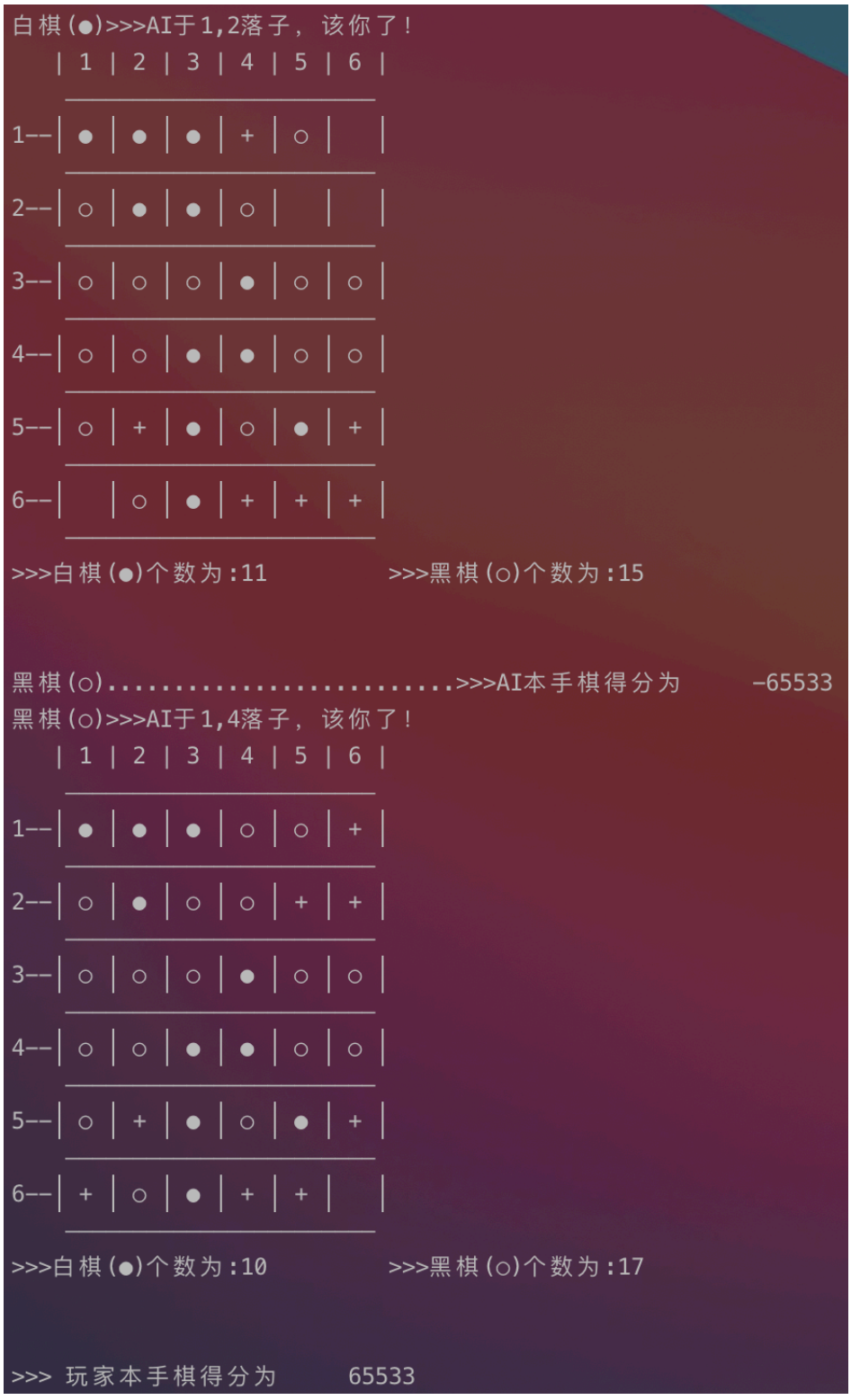

白棋(●)>>>AI于5,3落子，该你了！
| 1 | 2 | 3 | 4 | 5 | 6 |
1--| | + | ● | + | ○ | |
2--| + | + | ● | ○ | + | |
3--| + | ● | ● | ● | ● | ○ |
4--| ○ | ○ | ● | ● | ○ | + |
5--| | + | ● | ○ | | |
6--| | | | + | | |
>>>白棋(●)个数为:9
>>>黑棋(○)个数为:7
黑棋(○).....>>>AI本手棋得分为 -44
黑棋(○)>>>AI于4,6落子，该你了！
| 1 | 2 | 3 | 4 | 5 | 6 |
1--| | | ● | + | ○ | |
2--| | | ● | ○ | + | + |
3--| + | ● | ● | ● | ○ | ○ |
4--| ○ | ○ | ● | ● | ○ | ○ |
5--| + | + | ● | ○ | + | + |
6--| | | | + | + | |
>>>白棋(●)个数为:8
>>>黑棋(○)个数为:9
>>> 玩家本手棋得分为 21982

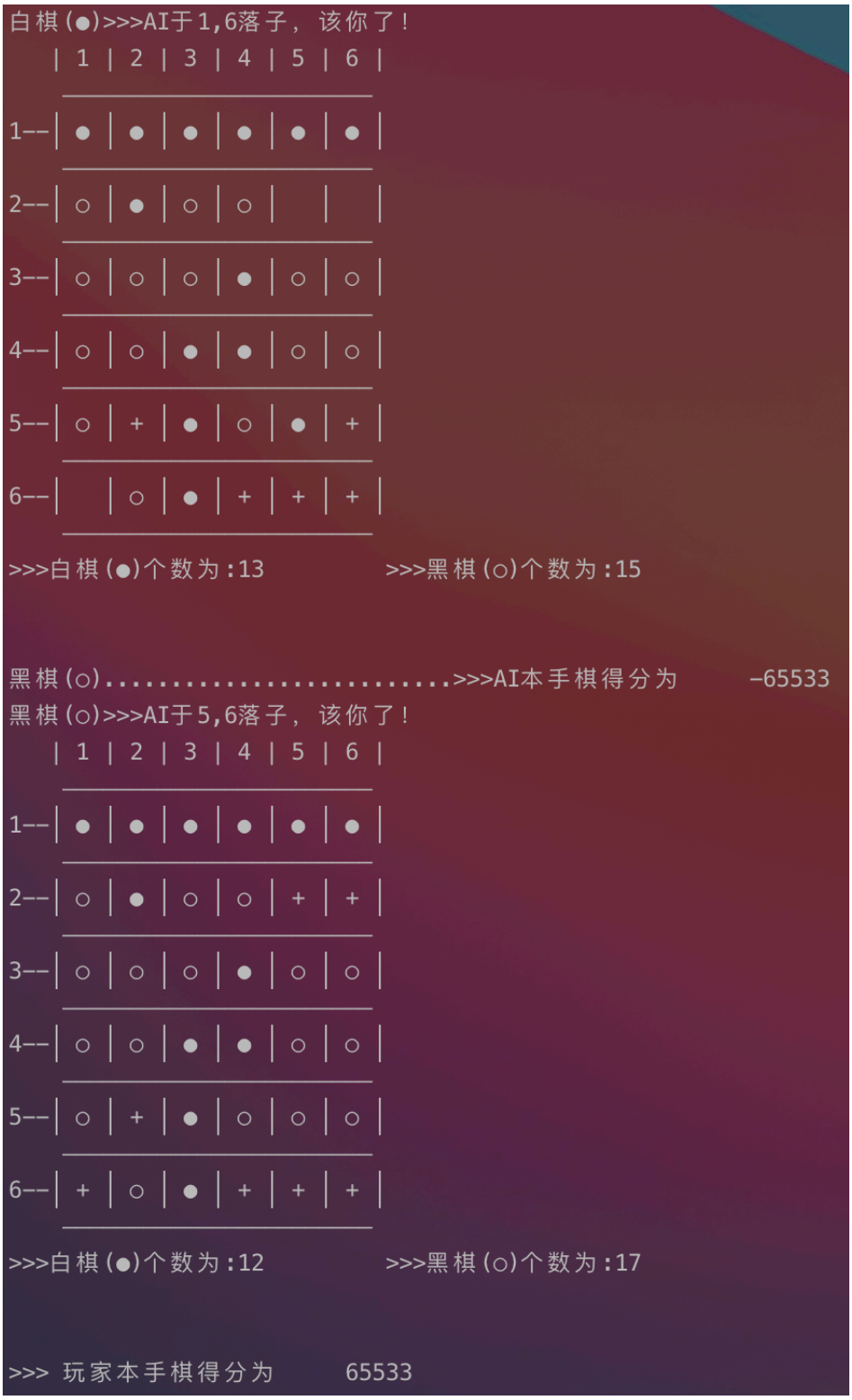

白棋(●)>>>AI于3,1落子，该你了！
| 1 | 2 | 3 | 4 | 5 | 6 |
1--| + | ● | + | ○ |
2--| + | + | ● | ○ |
3--| ● | ● | ● | ● | ○ | ○ |
4--| ○ | ● | ● | ● | ○ | ○ |
5--| + | + | ● | ○ |
6--| + |
>>>白棋(●)个数为:10
>>>黑棋(○)个数为:8
黑棋(○).....>>>AI本手棋得分为 -102
黑棋(○)>>>AI于6,2落子，该你了！
| 1 | 2 | 3 | 4 | 5 | 6 |
1--| ● | + | ○ |
2--| ● | ○ | + |
3--| ● | ● | ● | ● | ○ | ○ |
4--| ○ | ● | ● | ○ | ○ | ○ |
5--| + | ○ | ○ | + | + |
6--| ○ | + | + | + |
>>>白棋(●)个数为:8
>>>黑棋(○)个数为:11
>>> 玩家本手棋得分为 29976

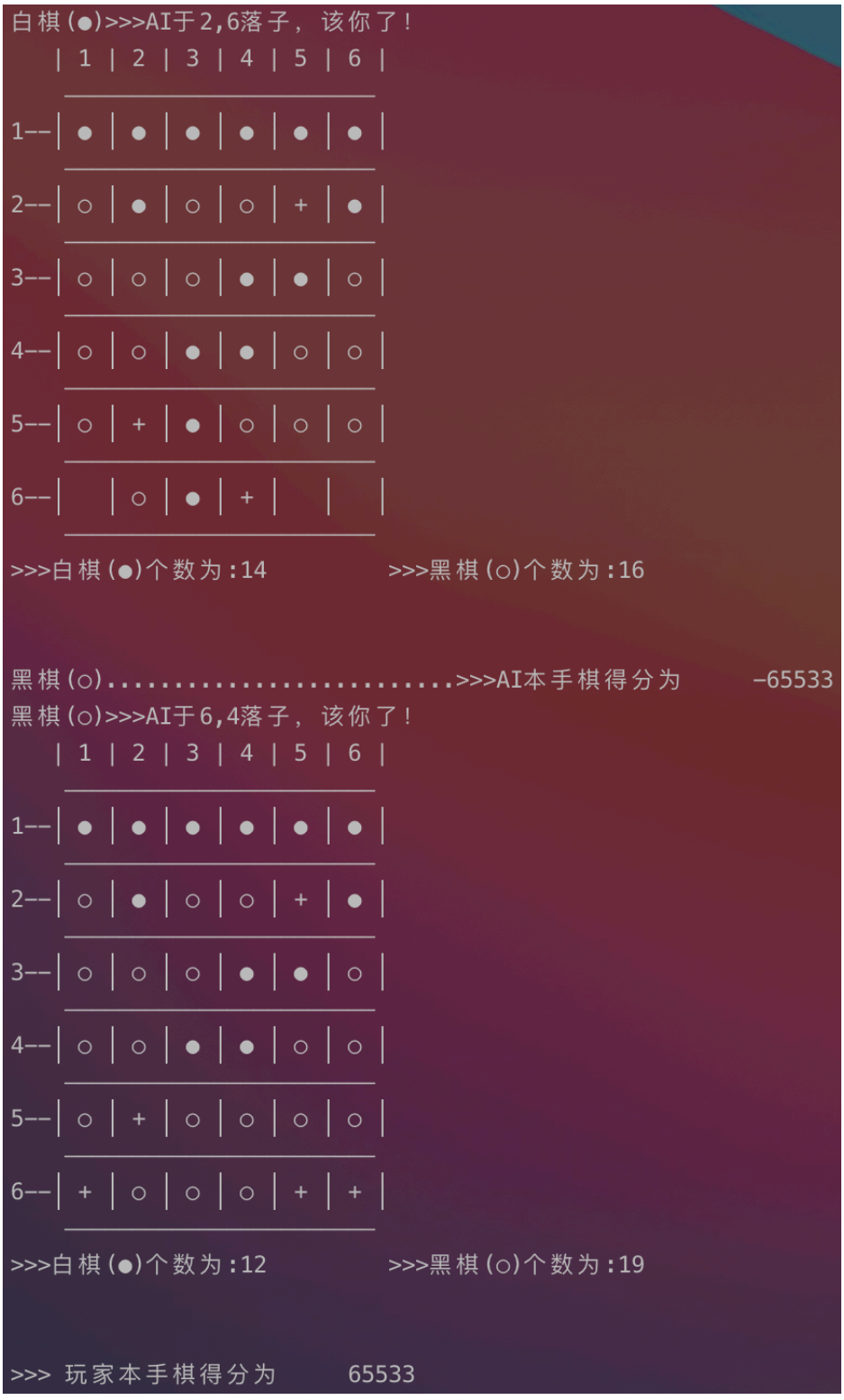
```
白棋(●)>>>AI于5,5落子，该你了！
   | 1 | 2 | 3 | 4 | 5 | 6 |
    ---------------------
1--|   | + | ● | + | ○ |   |
    ---------------------
2--| + | + | ● | ○ |   |   |
    ---------------------
3--| ● | ● | ● | ● | ○ | ○ |
    ---------------------
4--| ○ | ● | ● | ● | ○ | ○ |
    ---------------------
5--| + |   | ○ | ○ | ● | + |
    ---------------------
6--|   | ○ |   | + | + |   |
    ---------------------
>>>白棋(●)个数为:10          >>>黑棋(○)个数为:10


黑棋(○).........................>>>AI本手棋得分为      -151
黑棋(○)>>>AI于2,1落子，该你了！
   | 1 | 2 | 3 | 4 | 5 | 6 |
    ---------------------
1--|   |   | ● | + | ○ |   |
    ---------------------
2--| ○ | + | ● | ○ | + | + |
    ---------------------
3--| ○ | ○ | ● | ● | ○ | ○ |
    ---------------------
4--| ○ | ● | ○ | ● | ○ | ○ |
    ---------------------
5--|   | + | ○ | ○ | ● | + |
    ---------------------
6--|   | ○ | + | + |   |   |
    ---------------------
>>>白棋(●)个数为:7          >>>黑棋(○)个数为:14


>>> 玩家本手棋得分为      42304
```

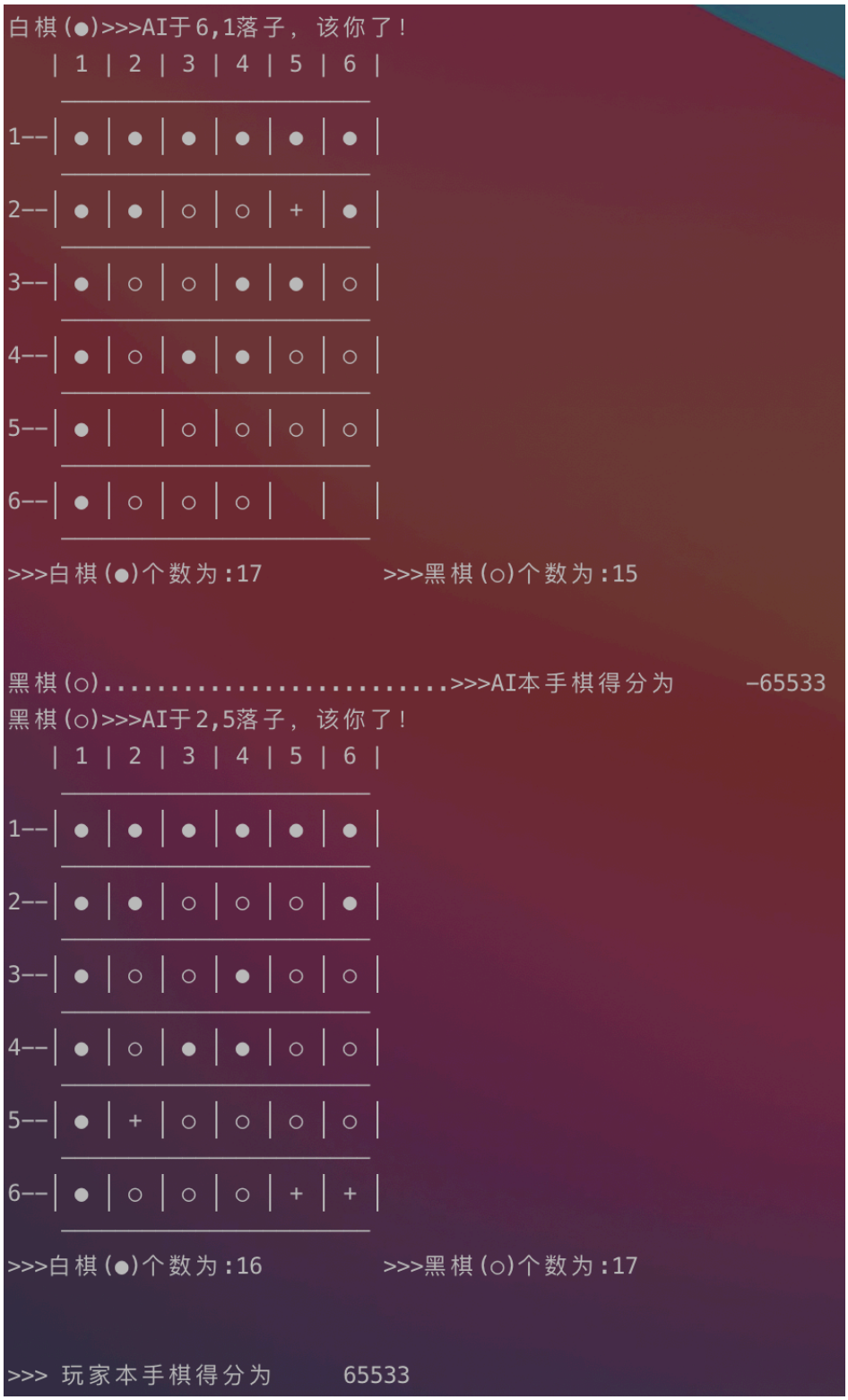

白棋(●)>>>AI于6,3落子，该你了！
| 1 | 2 | 3 | 4 | 5 | 6 |
1--| | + | ● | + | ○ | |
2--| ○ | + | ● | ○ | | |
3--| ○ | ○ | ● | ● | ○ | ○ |
4--| ○ | ● | ● | ● | ○ | ○ |
5--| + | + | ● | ○ | ● | + |
6--| | ○ | ● | + | + | |
>>>白棋(●)个数为:10
>>>黑棋(○)个数为:12
黑棋(○).....>>>AI本手棋得分为 -193
黑棋(○)>>>AI于2,2落子，该你了！
| 1 | 2 | 3 | 4 | 5 | 6 |
1--| + | + | ● | + | ○ | |
2--| ○ | ○ | ○ | ○ | + | + |
3--| ○ | ○ | ● | ● | ○ | ○ |
4--| ○ | ● | ● | ● | ○ | ○ |
5--| | | ● | ○ | ● | + |
6--| + | ○ | ● | + | + | |
>>>白棋(●)个数为:9
>>>黑棋(○)个数为:14
>>> 玩家本手棋得分为 52721

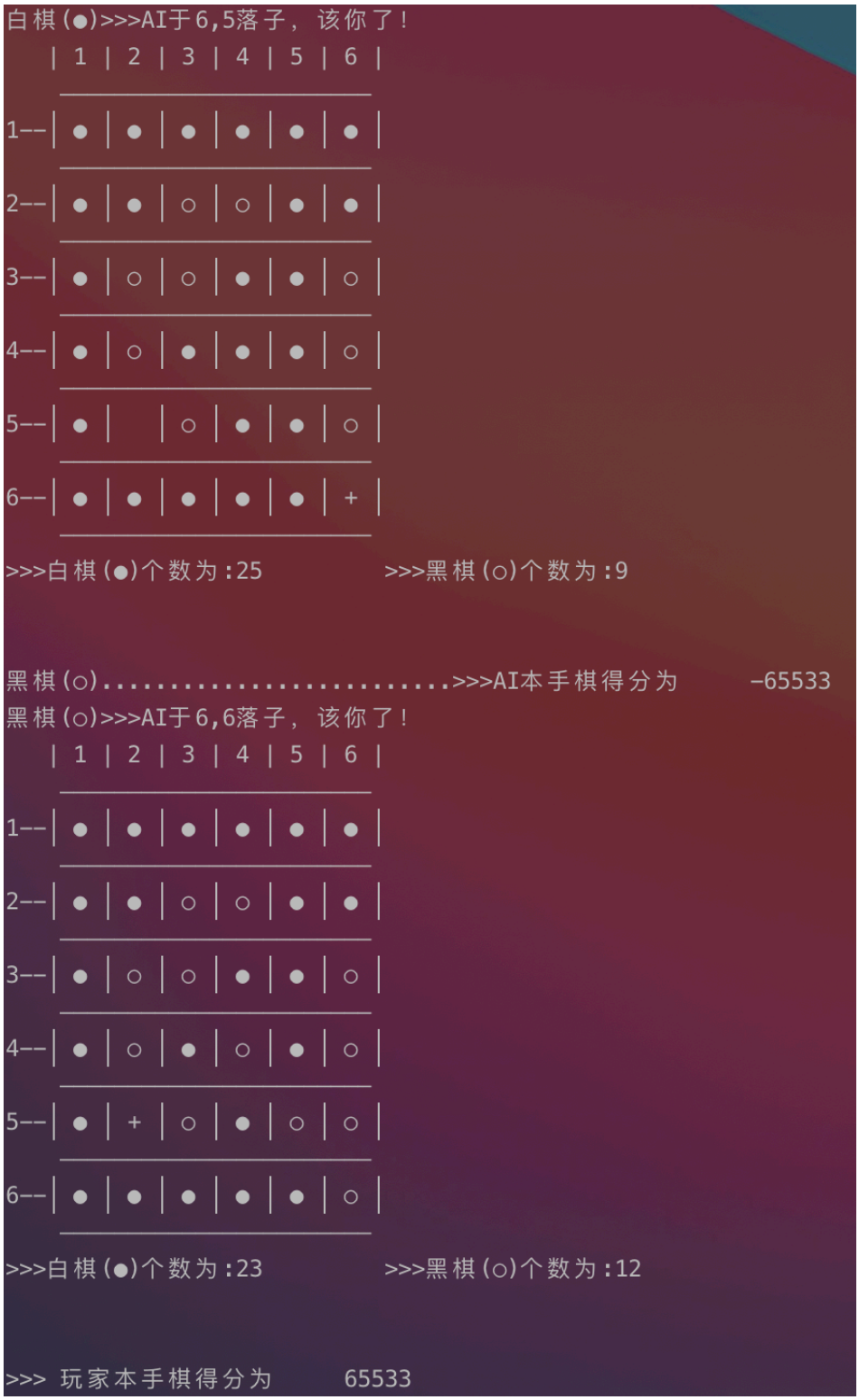

白棋(●)>>>AI于1,1落子，该你了！
| 1 | 2 | 3 | 4 | 5 | 6 |
1--| ● | + | ● | | ○ | |
2--| ○ | ● | ○ | ○ | | |
3--| ○ | ○ | ● | ● | ○ | ○ |
4--| ○ | ● | ● | ● | ○ | ○ |
5--| + | + | ● | ○ | ● | + |
6--| | ○ | ● | + | + | |
>>>白棋(●)个数为:11
>>>黑棋(○)个数为:13
黑棋(○).....>>>AI本手棋得分为 -214
黑棋(○)>>>AI于5,1落子，该你了！
| 1 | 2 | 3 | 4 | 5 | 6 |
1--| ● | + | ● | + | ○ | |
2--| ○ | ● | ○ | ○ | + | + |
3--| ○ | ○ | ○ | ● | ○ | ○ |
4--| ○ | ○ | ● | ● | ○ | ○ |
5--| ○ | + | ● | ○ | ● | + |
6--| + | ○ | ● | + | + | |
>>>白棋(●)个数为:9
>>>黑棋(○)个数为:16
>>> 玩家本手棋得分为 56514


白棋(●)>>>AI于1,2落子，该你了！
| 1 | 2 | 3 | 4 | 5 | 6 |
1--| ● | ● | ● | + | ○ | |
2--| ○ | ● | ● | ○ | | |
3--| ○ | ○ | ○ | ● | ○ | ○ |
4--| ○ | ○ | ● | ● | ○ | ○ |
5--| ○ | + | ● | ○ | ● | + |
6--| | ○ | ● | + | + | + |
>>>白棋(●)个数为:11
>>>黑棋(○)个数为:15
黑棋(○).....>>>AI本手棋得分为 -65533
黑棋(○)>>>AI于1,4落子，该你了！
| 1 | 2 | 3 | 4 | 5 | 6 |
1--| ● | ● | ● | ○ | ○ | + |
2--| ○ | ● | ○ | ○ | + | + |
3--| ○ | ○ | ○ | ● | ○ | ○ |
4--| ○ | ○ | ● | ● | ○ | ○ |
5--| ○ | + | ● | ○ | ● | + |
6--| + | ○ | ● | + | + | |
>>>白棋(●)个数为:10
>>>黑棋(○)个数为:17
>>> 玩家本手棋得分为 65533


白棋(●)>>>AI于1,6落子，该你了！
| 1 | 2 | 3 | 4 | 5 | 6 |
1--| ● | ● | ● | ● | ● | ● |
2--| ○ | ● | ○ | ○ |  |  |
3--| ○ | ○ | ○ | ● | ○ | ○ |
4--| ○ | ○ | ● | ● | ○ | ○ |
5--| ○ | + | ● | ○ | ● | + |
6--|  | ○ | ● | + | + | + |
>>>白棋(●)个数为:13
>>>黑棋(○)个数为:15
黑棋(○).....>>>AI本手棋得分为 -65533
黑棋(○)>>>AI于5,6落子，该你了！
| 1 | 2 | 3 | 4 | 5 | 6 |
1--| ● | ● | ● | ● | ● | ● |
2--| ○ | ● | ○ | ○ | + | + |
3--| ○ | ○ | ○ | ● | ○ | ○ |
4--| ○ | ○ | ● | ● | ○ | ○ |
5--| ○ | + | ● | ○ | ○ | ○ |
6--| + | ○ | ● | + | + | + |
>>>白棋(●)个数为:12
>>>黑棋(○)个数为:17
>>> 玩家本手棋得分为 65533


白棋(●)>>>AI于2,6落子，该你了！
| 1 | 2 | 3 | 4 | 5 | 6 |
1--| ● | ● | ● | ● | ● | ● |
2--| ○ | ● | ○ | ○ | + | ● |
3--| ○ | ○ | ○ | ● | ● | ○ |
4--| ○ | ○ | ● | ● | ○ | ○ |
5--| ○ | + | ● | ○ | ○ | ○ |
6--| | ○ | ● | + | | |
>>>白棋(●)个数为:14
>>>黑棋(○)个数为:16
黑棋(○).....>>>AI本手棋得分为 -65533
黑棋(○)>>>AI于6,4落子，该你了！
| 1 | 2 | 3 | 4 | 5 | 6 |
1--| ● | ● | ● | ● | ● | ● |
2--| ○ | ● | ○ | ○ | + | ● |
3--| ○ | ○ | ○ | ● | ● | ○ |
4--| ○ | ○ | ● | ● | ○ | ○ |
5--| ○ | + | ○ | ○ | ○ | ○ |
6--| + | ○ | ○ | ○ | + | + |
>>>白棋(●)个数为:12
>>>黑棋(○)个数为:19
>>> 玩家本手棋得分为 65533


白棋(●)>>>AI于6,1落子，该你了！
| 1 | 2 | 3 | 4 | 5 | 6 |
1--| ● | ● | ● | ● | ● | ● |
2--| ● | ● | ○ | ○ | + | ● |
3--| ● | ○ | ○ | ● | ● | ○ |
4--| ● | ○ | ● | ● | ○ | ○ |
5--| ● | | ○ | ○ | ○ | ○ |
6--| ● | ○ | ○ | ○ | | |
>>>白棋(●)个数为:17
>>>黑棋(○)个数为:15
黑棋(○).....>>>AI本手棋得分为 -65533
黑棋(○)>>>AI于2,5落子，该你了！
| 1 | 2 | 3 | 4 | 5 | 6 |
1--| ● | ● | ● | ● | ● | ● |
2--| ● | ● | ○ | ○ | ○ | ● |
3--| ● | ○ | ○ | ● | ○ | ○ |
4--| ● | ○ | ● | ● | ○ | ○ |
5--| ● | + | ○ | ○ | ○ | ○ |
6--| ● | ○ | ○ | ○ | + | + |
>>>白棋(●)个数为:16
>>>黑棋(○)个数为:17
>>> 玩家本手棋得分为 65533


白棋(●)>>>AI于6,5落子，该你了！
| 1 | 2 | 3 | 4 | 5 | 6 |
1--| ● | ● | ● | ● | ● | ● |
2--| ● | ● | ○ | ○ | ● | ● |
3--| ● | ○ | ○ | ● | ● | ○ |
4--| ● | ○ | ● | ● | ● | ○ |
5--| ● | | ○ | ● | ● | ○ |
6--| ● | ● | ● | ● | ● | + |
>>>白棋(●)个数为:25
>>>黑棋(○)个数为:9
黑棋(○).....>>>AI本手棋得分为 -65533
黑棋(○)>>>AI于6,6落子，该你了！
| 1 | 2 | 3 | 4 | 5 | 6 |
1--| ● | ● | ● | ● | ● | ● |
2--| ● | ● | ○ | ○ | ● | ● |
3--| ● | ○ | ○ | ● | ● | ○ |
4--| ● | ○ | ● | ○ | ● | ○ |
5--| ● | + | ○ | ● | ○ | ○ |
6--| ● | ● | ● | ● | ● | ○ |
>>>白棋(●)个数为:23
>>>黑棋(○)个数为:12
>>> 玩家本手棋得分为 65533


白棋(●)>>>AI于5,2落子，该你了！
| 1 | 2 | 3 | 4 | 5 | 6 |
1--| ● | ● | ● | ● | ● | ● |
2--| ● | ● | ○ | ○ | ● | ● |
3--| ● | ● | ○ | ● | ● | ○ |
4--| ● | ● | ● | ○ | ● | ○ |
5--| ● | ● | ● | ● | ○ | ○ |
6--| ● | ● | ● | ● | ● | ○ |
>>>白棋(●)个数为:27
>>>黑棋(○)个数为:9
白棋(●)胜
GAME OVER!
>>>>>>>>>>>>>>>>Again?(Y,N)<<<<<<<<<<<<<<<<
YES
or
NO
NO

程序结束，白棋胜