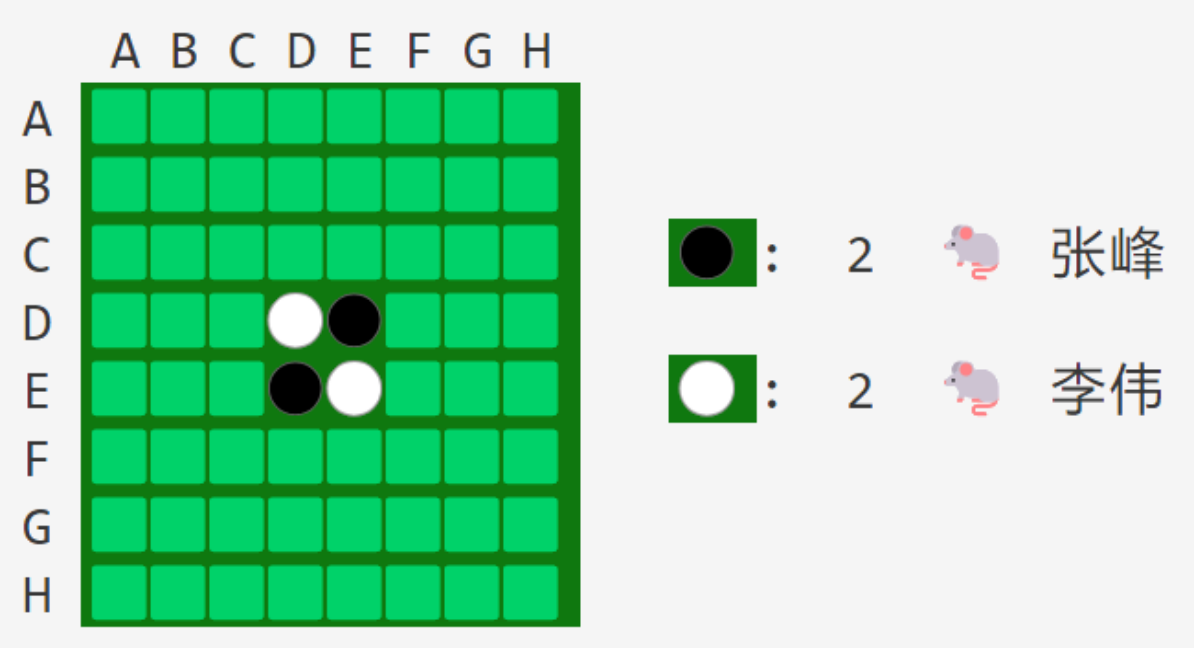
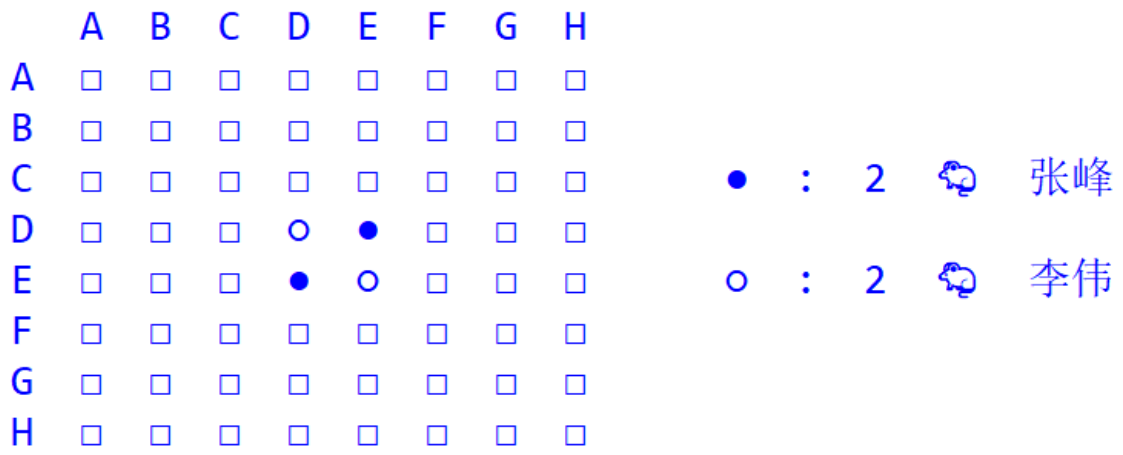
Python程序设计大作业：黑白棋

(ver.79 2024/11/19)

黑白棋[1]，也称翻转棋（Reversi）、奥赛罗棋（Othello）或正反棋（Anti reversi），是一种通过翻转对方棋子来决定胜负的策略游戏。
游戏在一个8x8的棋盘上进行，棋子需放在方格中间，下图是通过emoji字符绘制的黑白棋的示意图。棋盘右边的数值给出了相应玩家目前在棋盘上的棋子个数。



如果采用集成开发环境IDLE，相应的黑白棋的示意图如下所示：



在开始之前，建议先试玩一下在线版本的黑白棋游戏[2]

1 黑白棋

1.1 规则

1）**初始布局**：棋盘正中央四个位置已经放置了棋子，对于8x8的棋盘，黑棋位于DE 和 ED，白棋位于DD 和 EE。
2）**先行顺序**：黑方先行，双方交替下棋。
3）**合法棋步**：落子时必须在一个空格放置一个棋子，并翻转对手一个或多个棋子。
4）**翻转规则**：新落下的棋子与已有同色棋子之间夹住的所有对手棋子必须翻转。夹住的棋子可以是横向、纵向或斜向，且夹住的位置必须全部是对手的棋子。
5）**多方向翻转**：一步棋可以在多个方向上翻转所有被夹住的棋子，不能选择不翻转某个棋子。

6）**落子条件**：必须至少翻转一颗对手的棋子才能落子。如果一方没有合法落子，则该方弃权，由对手继续落子。
7）**强制落子**：如果有合法落子，必须落子，不得弃权。
8）**游戏结束**：游戏持续进行，直到棋盘填满或双方都无合法落子。

1.2 流程分析

在下面的分析中，以下名词有特殊含义：

- board：棋盘；
- piece：棋子类型，包括黑棋(BLACK)、白棋(WHITE)和无棋(EMPTY)；实际上是棋盘上放置棋子的状态；
- player：玩家，
- name：玩家名称。

1）游戏流程
分析黑白棋游戏，游戏流程大致如下：

- 创建两个玩家
- 开始游戏
    - 初始化棋盘；
    - 显示棋盘；
    - 其他必要的初始化；
    - 主循环
        - 玩家1走一步；
        - 玩家2走一步；
    - 输出黑白棋结果。

上述流程大致可以使用下面的代码来实现

```
def move_one_step(player, board):
    name, piece, move = player['name'], player['piece'], player['move']

    row, col = move(board, piece, name)
    if ! is_valid_move(board, row, col, piece):
        return WRONG

    do_move(board, row, col, piece)
    display_board(board)

    return status

def game(player1, player2):
    board = init_board()
    display_board(board)
    while not finished:
        status = move_one_step(player1, board)
        check(status)
        status = move_one_step(player2, board)
        check(status)
    output(...)
```

```
22
23  def main():
24      player1, player2 = create_player('A',move_input), create_player('B',
    move_random)
25      game(player1, player2)
```

2）相关函数功能说明
2.1）move_xxx(board, piece, name)函数
该函数返回玩家的下一个位置。通过实现不同的 move_xxx 函数，可以实现自动放置棋子或手动输入棋子。所有的 move_xxx 函数具有相同的参数，并返回放置棋子的位置 (row, col)。玩家放置棋子的方法包括：

- 用户输入行和列，并给予提示；
- 从当前所有可放置棋子的位置中随机选择一个；
- 使用AI算法计算出最佳的位置。

2.2）create_player(name, move)函数
该函数用于创建玩家。创建玩家时，需要指定玩家名称(name)和玩家指定的move函数。通过不同的move_xxx函数的组合，可以支持：

- 两个用户之间的游戏
- 一个用户和采用某种策略的机器玩家之间的游戏
- 采用不同策略的两个机器玩家之间的游戏

2.3）move_one_step(player, board)函数
该函数用于处理玩家下一手棋的操作。通过调用玩家的 move 函数，获取放置棋子的位置(row, col)；然后将棋子放置在棋盘上，并翻转符合条件的所有棋子。在翻转完成后，需要调用 display_board(board) 函数来显示更新后的棋盘状态。
函数的返回值是当前游戏的状态，包括是否出错、是否无棋可下等。

2.4）game(player1, player2, n = 8)函数
该函数是游戏的入口。需要指定两个玩家 player1 和 player2。在函数内部，需要初始化棋盘并显示棋盘；然后开始循环，让玩家1和玩家2依次下一手棋。每下一手棋后，都需要检查当前的游戏状态，以确定是否需要结束游戏。

2.5）init_board(n=8)函数
该函数创建并初始化棋盘。

2.6）display_board(board)函数
该函数显示棋盘的状态。

上述函数中，init_board、display_board、create_player等给出了初步实现。

2 作业要求

1）提供的附件
提供othellolib.py文件、run.py文件。

2）代码要求
实现上面规定的函数以及其他必要的函数，实现黑白棋功能，且要求：
2.1）move()函数

- 实现move_input()函数，让用户输入指定的位置，并给予用户提示。具体地，当用户输入?时，提示所有可以放置棋子的位置；当用户输入位置错误或者输入了不能放置棋子的位置，需要提示用户重新输入；

- 实现move_random()函数，随机从可放置棋子的位置中选择一个；
- 可选要求：可以参考[2],[3]，实现一个具有智能的move_AI()函数。

2.2）display_board函数
把棋盘输出的形式更改为文档中othello示意图的形式。

3）提交要求
提交othellolib.py文件、run.py文件和必要的项目说明。

3 参考资料

[1] 黑白棋，wiki. https://zh.wikipedia.org/wiki/%E9%BB%91%E7%99%BD%E6%A3%8B
[2] 黑白棋在线游戏. https://www.heibaiqi123.com/
[3] 黑白棋技巧. https://www.zhihu.com/question/25271618