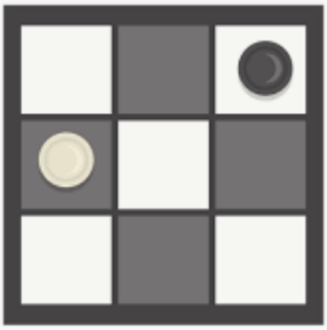
Code / Project : CME1251 / 3
Year / Semester : 2021-2022 Fall Semester
Duration : 5 weeks



Project: CENG Checkers (or Cengers)

The aim of the project is to develop a two-player checkers-like board game called Ceng Checkers.

General Information

The game is played on a 8*8 board. Players of the game are human (x) and computer (o). Human player starts the game. The game is turn based. The goal of the game is to be the first player to move all 9 pieces across the board and into their own home area. Each player's home area is the opposing 3*3 area.

Initial Board

At the beginning of the game, each player has 9 pieces (as 3*3). Human player has x pieces in the bottom right 3*3 area of the board and computer player has o pieces in the top left 3*3 area. Each player wants to move his/her pieces to their home (opposite side of the board (3*3 area)).

Game Playing Rules

All the moves are in 4 directions, diagonal moves cannot be used.

There are 2 move types for a player in each turn: Either a step or jump(s).

- **Step**: If adjacent square of a piece in any direction (left, right, up or down) is empty, that piece can step into the empty square.
- **Jump**: A piece can jump over only a single adjacent piece (his/her or opponent's). Jumping over 2 or more pieces or distant pieces is not allowed.

Jumping operations can be continued with successive jumps (called jump chain) if possible, in the same turn. On the contrary, step operation is a single one. There is no capturing in this game, so all pieces remain active during the game.

For each step or jump operation; human user firstly chooses the piece to move, then chooses the target square of the piece.

1. Move cursor to the location of the piece
2. Choose the piece by pressing key Z
3. Move cursor to the target location
4. Choose target square by pressing key X
   After the move;
5. If there is no successive jumps, end the move by pressing key C
6. If the player wants to jump again, go to stage 3

Sample Game Screens

```
  12345678
 +--------+   Round: 1
1|ooo.....|   Turn : x
2|ooo.....|
3|ooo.....|
4|........|
5|........|
6|.....xxx|
7|.....xxx|
8|.....xxx|
 +--------+

  12345678
 +--------+   Round: 1
1|ooo.....|   Turn : o
2|ooo.....|
3|ooo.....|
4|........|
5|.....x..|
6|......xx|
7|.....xxx|
8|.....xxx|
 +--------+
```

```
      .
      .
      .

  12345678
 +--------+   Round: 27
1|xx......|   Turn : o
2|xxx.....|
3|.x......|
4|x.......|
5|x.......|
6|...o.oxo|
7|.....ooo|
8|.....ooo|
 +--------+
  12345678
 +--------+   Round: 2
1|ooo.....|   Turn : x
2|oo......|
3|ooo.....|
4|..o.....|
5|.....x..|
6|......xx|
7|.....xxx|
8|.....xxx|
 +--------+

  12345678
 +--------+   Round: 2
1|ooo.....|   Turn : o
2|oo......|
3|ooo.....|
4|..o.....|
5|.....xx.|
6|.......x|
7|.....xxx|
8|.....xxx|
 +--------+

  12345678
 +--------+   Round: 3
1|ooo.....|   Turn : x
2|o.......|
3|ooo.....|
4|..oo....|
5|.....xx.|
6|.......x|
7|.....xxx|
8|.....xxx|
 +--------+

  12345678
 +--------+   Round: 3
1|ooo.....|   Turn : o
2|o.......|
3|ooo.....|
4|..oo....|
5|....xx..|
6|.......x|
7|.....xxx|
8|.....xxx|
 +--------+

  12345678
 +--------+   Round: 4
1|ooo.....|   Turn : x
2|o.......|
3|ooo.....|
4|...o....|
5|....xx..|
6|....o..x|
7|.....xxx|
8|.....xxx|
 +--------+

  12345678
 +--------+   Round: 4
1|ooo.....|   Turn : o
2|o.......|
3|ooox....|
4|...o....|
5|....x...|
6|....o..x|
7|.....xxx|
8|.....xxx|
 +--------+

  12345678
 +--------+   Round: 5
1|ooo.....|   Turn : x
2|o.......|
3|oo.xo...|
4|...o....|
5|....x...|
6|....o..x|
7|.....xxx|
8|.....xxx|
 +--------+
```

```
  12345678
 +--------+   Round: 28
1|xx......|   Turn : x
2|xxx.....|
3|.x......|
4|x.......|
5|x.......|
6|....ooxo|
7|.....ooo|
8|.....ooo|
 +--------+

  12345678
 +--------+   Round: 28
1|xxx.....|   Turn : o
2|xxx.....|
3|.x......|
4|x.......|
5|........|
6|....ooxo|
7|.....ooo|
8|.....ooo|
 +--------+

  12345678
 +--------+   Round: 29
1|xxx.....|   Turn : x
2|xxx.....|
3|.x......|
4|x.......|
5|.......o|
6|....oox.|
7|.....ooo|
8|.....ooo|
 +--------+

  12345678
 +--------+   Round: 29
1|xxx.....|   Turn : o
2|xxx.....|
3|xx......|
4|........|
5|.......o|
6|....oox.|
7|.....ooo|
8|.....ooo|
 +--------+

  12345678
 +--------+   Round: 30
1|xxx.....|   Turn : x
2|xxx.....|
3|xx......|
4|........|
5|........|
6|....ooxo|
7|.....ooo|
8|.....ooo|
 +--------+

  12345678
 +--------+   Round: 30
1|xxx.....|   Turn : o
2|xxx.....|
3|xx......|
4|........|
5|......x.|
6|....oo.o|
7|.....ooo|
8|.....ooo|
 +--------+

  12345678
 +--------+   Winner: o
1|xxx.....|
2|xxx.....|
3|xx......|
4|........|
5|......x.|
6|.....ooo|
7|.....ooo|
8|.....ooo|
 +--------+
```

Suggested Weekly Program

1. Discussing and designing solution alternatives. Creating the necessary variables, structures. Screen. Designing functions. Planning task distribution and scheduling.
2. For human: Cursor moves. Choosing piece. Choosing target. Step operation.
3. For computer: Step operation.
4. For computer: Single jump operation. For human: Single jump operation and successive jump operations.
5. For computer: Double jump operation. Remaining parts of the game. Testing/Debugging.

First Evaluation: 24.12.2021 **Report:** 24.12.2021	**Final Evaluation:** 7.1.2022 (powerpoint + poster) **Report:** 7.1.2022