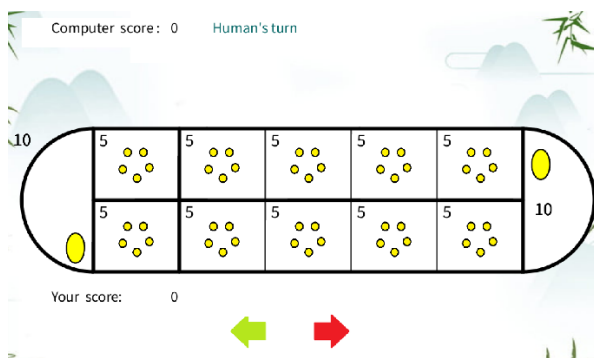# PROJECT README

Ngoc Thao Tran

Matrikelnummer : 5293553

O an quan game

"O an quan" is a traditional Vietnamese board game, which is commonly played by children.

Game's setup

Game's board

The game is set up like above.

The game's board is drawn with a rectangle which is divided into 10 squares, each side has 5 symmetrical fields, which contain small stones. On the two sides of the rectangle are two semicircles which contain big stone. The square fields are called **folk fields** and the two semi-circular fields are called **king fields**.

Game's figures

Stones:

- Small stone = 1 point

- Big stone = 10 small stones = 10 point

Game's players

Number of players is 2. Each player plays on 1 side of the board and is allowed to play only with the stones on the fields of their side.

Game's rules

Goal of the game

The winner is the one who wins more stones (points) at the end of the game.

Rules of scattering and winning stones

- Each player in their turn moves the game's stones to take as many stones for themselves as possible.
- When it is the player's turn, the player uses all the stones of one square on his/her side to scatter. The stones can be scattered clockwise or counterclockwise (left ⬅ or right ➡).
- In case there are no stones left on player's side to choose from, 5 stones from player's resources will be placed on the fields with 1 stone each field.

- When spreading the last piece, the players proceed as follows depending on the situation:
  - **Continuing scattering**: If the next field has stones, and is not king's field, then use all these stones to continue scattering in the chosen direction.*(5),(6)*
  - **Gaining stones**: If the next field is an empty field followed by a field with stones, then the player can take all the stones in this field, and further fields which are separated by empty fields. *(7)*
  -
    Human player plays on the first position from the left, in the left direction, and gains 6 stones .
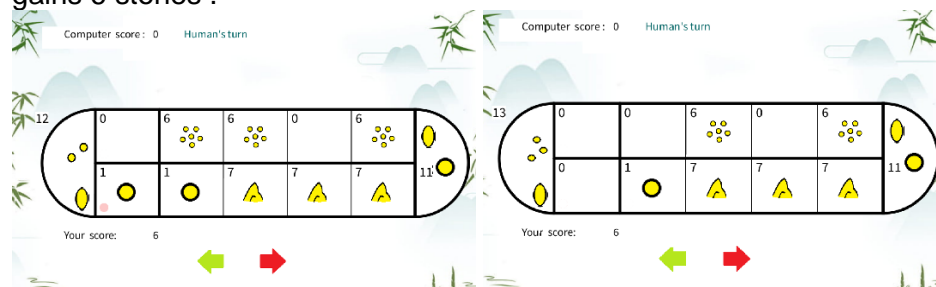
  - **Losing turn**:
  - If the next field is king field. For example: human player plays stones from the first position from the left, in the right direction.
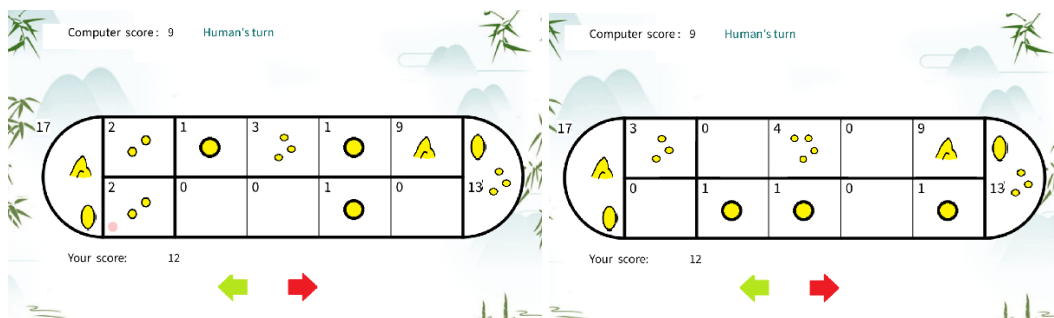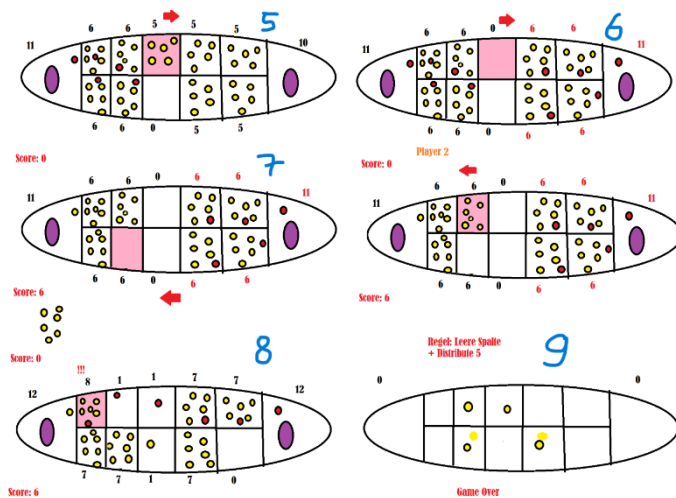

    At the end, the stone scattering must stop in front of king field.

  - or next fields are >=2 empty fields, or player has won some stones, then player loses his turn. *(8)*

Game over

The game ends when: *(9)*

- There are no more stones in both king fields *or:*
- When a player does not have enough resources to place stones on the board to play.

Picture to clarify the game's rules.

Run the game with Processing

Run MainServer class and MainClient class to run the program. Program only starts when both threads run. Notice the following:

- Human's turn is first.
- Clicking on the **direction** and **then the stone field** of choice to take turn.
- Play when "Human's turn" appears on the screen. Otherwise, when it is computer's turn, "Computer's turn" text appears on the screen.
- Computer will play automatic after human's turn.

Run the game on Jshell

Game is played automatic by computers.

```
jshell> game = Game.newGame()
[10, 5, 5, 5, 5, 5, 10, 5, 5, 5, 5, 5]
game ==> model.Game@ba8a1dc

jshell> while(!game.isEndgame()){
   ...>     game = game.play(game.randomMove());
   ...>     System.out.println(game);}
Position :8
Direction:-1
Stonespicked:5
[11, 6, 0, 6, 6, 6, 11, 0, 0, 6, 6, 6]
player:2
Player 2 score: 6
Player 1 score: 0
Score turn :6
model.Game@504bae78
Position :1
Direction:-1
Stonespicked:6
[12, 0, 0, 6, 6, 6, 11, 1, 1, 7, 7, 7]
player:1
Player 2 score: 6
Player 1 score: 0
Score turn :0
model.Game@3b764bce
Position :8
Direction:-1
Stonespicked:1
[12, 0, 0, 6, 6, 6, 11, 2, 0, 7, 7, 7]
player:2
Player 2 score: 6
Player 1 score: 0
Score turn :0
model.Game@759ebb3d
Position :3
```

Run game of human versus computer. Player 2: human, Player 1: computer. Human plays first. Computer plays random move.

```
jshell> var game = Game.of(new int[]{10,5,5,5,5,5,10,5,5,5,5,5}, new ArrayList<>(), 0, 0, 2)
game ==> Board: [10, 5, 5, 5, 5, 5, 10, 5, 5, 5, 5, 5]

jshell> game = game.play(Move.of(10,1))
[11, 6, 6, 6, 0, 6, 11, 6, 6, 6, 0, 0]
player:2
Player 2 score: 6
Player 1 score: 0
Score turn :6
game ==> Board: [11, 6, 6, 6, 0, 6, 11, 6, 6, 6, 0, 0]

jshell> game = game.play(game.randomMove())
Position :5
Direction:-1
Stonespicked:6
[12, 7, 7, 7, 1, 0, 11, 6, 6, 0, 0, 1]
player:1
Player 2 score: 6
Player 1 score: 6
Score turn :6
game ==> Board: [12, 7, 7, 7, 1, 0, 11, 6, 6, 0, 0, 1]

jshell> game = game.play(Move.of(9,1))
This field has no stones to play
game ==> Board: [12, 7, 7, 7, 1, 0, 11, 6, 6, 0, 0, 1]

jshell> game = game.play(Move.of(10,1))
This field has no stones to play
game ==> Board: [12, 7, 7, 7, 1, 0, 11, 6, 6, 0, 0, 1]

jshell> game = game.play(Move.of(11,1))
[13, 0, 8, 8, 2, 1, 12, 7, 7, 0, 0, 0]
player:2
Player 2 score: 6
Player 1 score: 6
```

If player chooses king field or empty field to play, game will give message for player to choose again.

```
jshell> game = game.play(Move.of(11,1))
[13, 0, 8, 8, 2, 1, 12, 7, 7, 0, 0, 0]
player:2
Player 2 score: 6
Player 1 score: 6
Score turn :0
game ==> Board: [13, 0, 8, 8, 2, 1, 12, 7, 7, 0, 0, 0]

jshell> game = game.play(game.randomMove())
Position :3
Direction:-1
Stonespicked:8
[14, 1, 9, 0, 2, 1, 12, 8, 8, 1, 1, 1]
player:1
Player 2 score: 6
Player 1 score: 6
Score turn :0
game ==> Board: [14, 1, 9, 0, 2, 1, 12, 8, 8, 1, 1, 1]

jshell> game = game.play(Move.of(6,1))
It is not allowed to play king fields.
game ==> Board: [14, 1, 9, 0, 2, 1, 12, 8, 8, 1, 1, 1]

jshell> game = game.play(Move.of(0,1))
It is not allowed to play king fields.
game ==> Board: [14, 1, 9, 0, 2, 1, 12, 8, 8, 1, 1, 1]
```

Language

Programming language: Java JDK 19 download

Library

- Processing Processing download

Sources

[1] T. T. Hoang, M. T. Dong, and X. V. Nguyen, 'Thesis report O an quan game'. Jul. 2016. [Online].(Available: https://123docz.net/document/3974969-bao-cao-mon-tri-tue-nhan-tao-game-o-an-quan.htm)

[2] Prof. M. Weigel 'Lecture codes: VW10 - Pong.java' . 2022