

## The Halma game

The Halma game is worldwide-popular board game with the setups and rules as follows [1]. In this course project assignment, you are required to program such game.

### Setup

- The board consists of a grid of  $16 \times 16$  squares (as in Figure 1).
  - Each player's camp consists of a cluster of adjacent squares in one corner of the board. These camps are delineated on the board (as in Figure 2).
    - For two-player games, each player's camp is a cluster of 19 squares. The camps are in opposite corners.
    - For four-player games, each player's camp is a cluster of 13 squares. Each of the four corners of the board is a camp.
  - Each player has a set of pieces in a distinct color, of the same number as squares in each camp.
- The game starts with each player's camp filled by pieces of their own color.

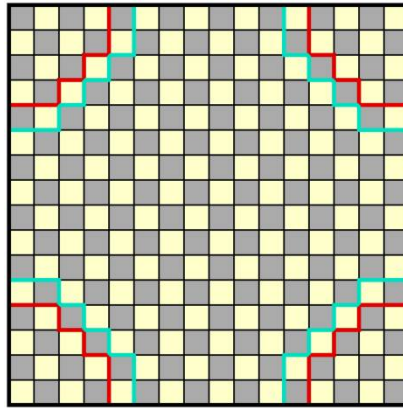


Fig 1. The board

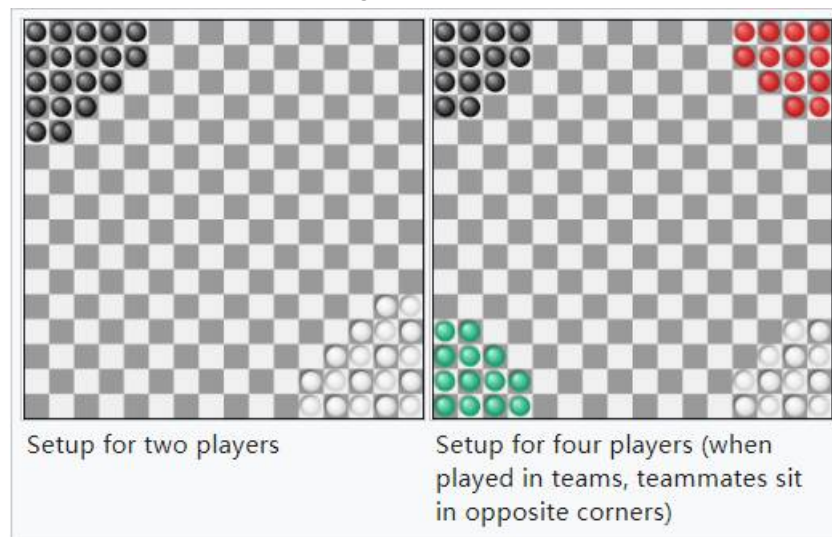


Fig 2. Setups of Halma

### The Rules

- Players randomly determine who will move first.
- Pieces can move in eight possible directions (orthogonally and diagonally).

- Each player's turn consists of moving a single piece of one's own color in one of the following plays (as in Figure 3):
  - One move to an empty square:
    - Place the piece in an empty adjacent square.
    - This move ends the play.
  - One or more jumps over adjacent pieces:
    - An adjacent piece of any color can be jumped if there is an adjacent empty square on the directly opposite side of that piece.
    - Place the piece in the empty square on the opposite side of the jumped piece.
    - The piece that was jumped over is unaffected and remains on the board.
    - After any jump, one may make further jumps using the same piece, or end the play.
- Once a piece has reached the opposing camp, a play cannot result in that piece leaving the camp.
- If the current play results in having every square of the opposing camp occupied by one's own pieces, the acting player wins. Otherwise, play proceeds clockwise around the board.

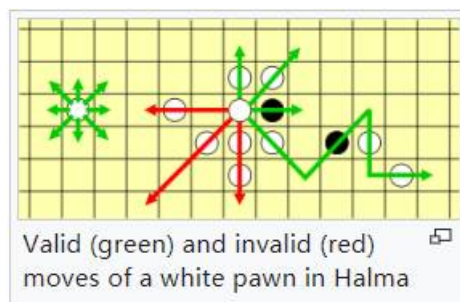


Fig 3. The moving rules of Halma pieces

## Project Requirement

You are required to form groups of two or three. Please design and implement a Java program to simulate the Halma game for two-four players. Note that the only programming language you can use in this project is Java. (Scala and Kotlin can also be accepted)

There are four tasks below to accomplish, each of which has several points towards the final mark of a group. A framework/skeleton of the game will be released shortly to facilitate your programming.

Please kindly note that though you work as a team, each of the team members would be graded by the individual contribution to the project. Normally the member who makes more contributions to the project would be graded higher than others.

For the same quality of work, the group of three members would be graded 90% of the group of two members.

### Task 1: Initialize the Halma Game (15 points)

- Your program should be able to initialize a new Halma game, which includes all lines and all Halma components with correct position in your Halma board.
- Your program should be able to display the status of the game (In Progress, e.g., Player Turn,

two-player games or four-player).

- Your program should be able to initialize two types of game including two-player games or four-player game

### **Task 2: Load and save a Halma game (15 points)**

- Your program should be able to load an existing Halma game from a text file with a pre-defined format by clicking a button. After loading, all the pieces should be placed at their positions given in the text file.
- Your program should be able to perform error check, e.g., the number of the pieces should maintain consistent, etc.
- Your program should be able to save the current Halma game to a text file. Only the position of pieces and turn of player need to be recorded.

### **Task 3: Play the Halma game (30 points)**

- Your program should let the player to move all Halma pieces following the movement rules described above.
- Your program should detect the winning status of the game, and end the game when there is a winner.

### **Task 4: Graphical User Interface (20 points)**

- Your program should have a graphical user interface using Java Swing. (FX can also be accepted)

### **Bonus (20 points)**

If your program satisfies all the above basic requirements, you will get 80 points. The remaining 20 points will be given as bonus. You are highly encouraged to go beyond our requirements. Below are some possible ways to get bonus. Compared to the bonus points, the basic points are easier to get. **Here you need to rely on your own ability to present your programming charm!**

- Design Human vs. Machine Mode, and make the machine player smarter.
- Design a platform for your game, such as adding multi-user (more than 4, that means you need to design a more complex board), multi-loading, ranking list, adding start menu for selecting the game modes, etc.
- Make your game looks nicer, such as changing the theme, adding sound effect, adding background music, adding more prompt label when playing the game.
- Show possible moves when a Halma piece is selected.
- Show some types of warning when there is about to be a winner.
- Play animations of Halma movements from one place to another, play the process of moving Halma pieces after loading the “moving steps file”, playback the process of the battle.
- Undo operation.
- Support on-line mode in Local Area Network.
- And more.

### **Reference:**

[1] <https://en.wikipedia.org/wiki/Halma#Rules>.