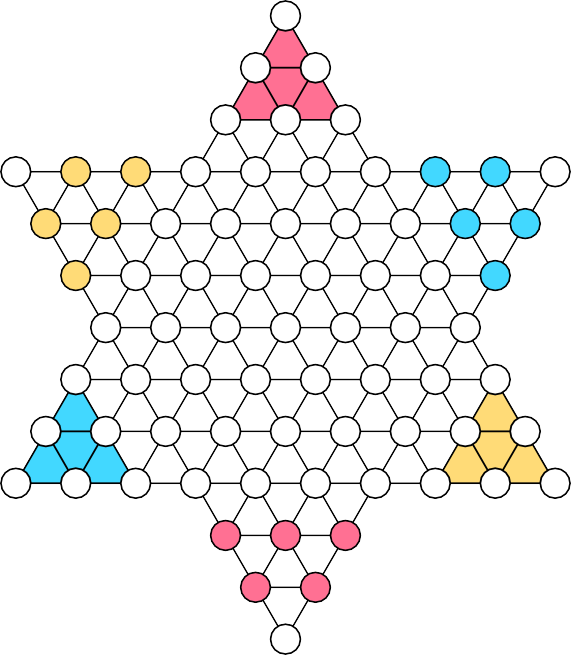
Chinese checkers

# Task Summary

Implement an agent for a variant of the game Sternhalma/Chinese checkers.



**C C**

**B B**

**C C**

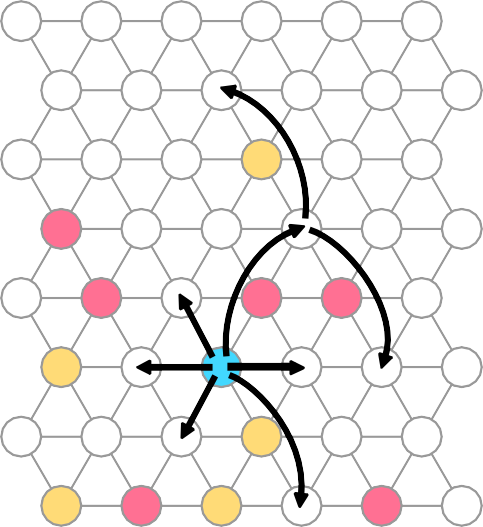
**B B**

**C**

**B**

**A A A**

**A A**



**C**

**A**

**A**

**A**

**A**

**C**

**B**

**C**

**C**

**A**

**C**

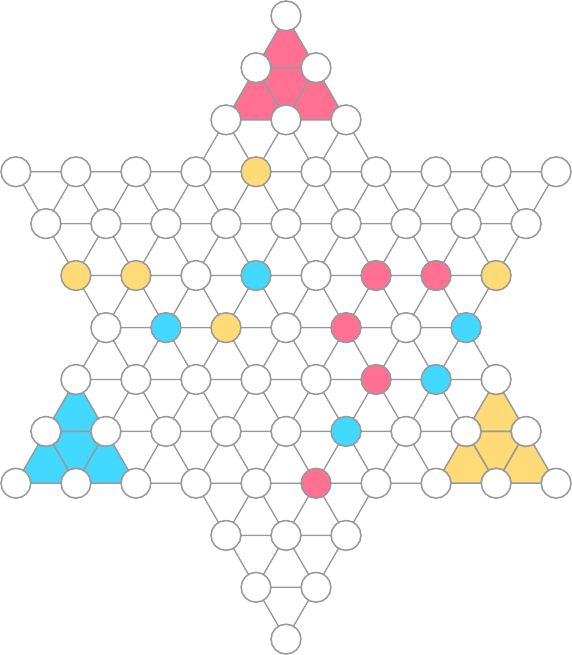
**A**

Figure 1: Starting position for a game with players *A*, *B* and *C* (left) and an illustration of the move rules (right).Didactic objectives

# Rules of FAUhalma

**FAUhalma** is a 3-player game, modified from the original game Sternhalma/Chinese check- ers [[CC]](#_bookmark5) for AISysProj. Like the original, FAUhalma is played on a star-shaped board. Each player has several **pegs**, which originally reside in the player’s **starting corner**. The goal of each player is to move their pegs into the opposite corner (the **home**) as quickly as possible. Figure [1](#_bookmark0) illustrates the starting position and the homes (shaded areas). Players take turns counterclockwise. When it is their turn, a player moves one of their pegs by either moving it to an adjacent space or by hopping over other pegs (see Figure [1).](#_bookmark0) Here are the move rules in more detail:

* **Simple move**: Move the peg to an adjacent empty space (in any direction).
* **Simple hop**: If there is a peg on an adjacent space *S*, the peg can “jump over it” as long as the space behind *S* is empty.
* **Hop chain**: A chain of simple hops; i.e. if a peg can hop from *S*1 to *S*2 and it could then hop from *S*2 to *S*3, then it can also hop from *S*1 to *S*3 in a single move. However, at the end of a chain, the peg must land on a different space than it started from. Note that a simple move cannot be combined with hops.
* **Swap rule**: Spaces in the home of the moving player that are occupied by an oppo- nent’s peg can also be considered empty for the rules above (except for the intermediate spaces in a hop chain). If a peg is moved from *S*1 to *S*2 and *S*2 is occupied by an oppo- nent’s peg, then the peg from *S*2 is moved to *S*1, i.e. the pegs are swapped. The swap



**(-3,6)**

**(-3,5) (-2,5)**

**(-3,4) (-2,4) (-1,4) C**

**(-6,3) (-5,3) (-4,3) (-3,3) (-2,3) (-1,3) (0,3) (1,3) (2,3) (3,3)**

**(-5,2) (-4,2) (-3,2) (-2,2) (-1,2) (0,2) (1,2) (2,2) (3,2) C C B A A C**

**(-4,1) (-3,1) (-2,1) (-1,1) (0,1) (1,1) (2,1) (3,1)**

**B C A B**

**(-3,0) (-2,0) (-1,0) (0,0) (1,0) (2,0) (3,0)**

**A B**

**(-3,-1) (-2,-1) (-1,-1) (0,-1) (1,-1) (2,-1) (3,-1) (4,-1)**

**B**

**(-3,-2) (-2,-2) (-1,-2) (0,-2) (1,-2) (2,-2) (3,-2) (4,-2) (5,-2)**

**A**

**(-3,-3) (-2,-3) (-1,-3) (0,-3) (1,-3) (2,-3) (3,-3) (4,-3) (5,-3) (6,-3)**

**(1,-4) (2,-4) (3,-4)**

**(2,-5) (3,-5)**

**(3,-6)**

Figure 2: Coordinate system for FAUhalma.

rule prevents players from blocking each other by moving pegs into their opponents’ homes.

It is theoretically possible that a player has no legal move. In that case the blocked player loses. The first player who has moved all their pegs into their home wins. The remaining players compete for the second place.

# Coordinates, Positions and Moves

To test your agents, we need to have a standardized way of sharing moves and positions, which will be discussed in this section.

## Coordinate System

We will refer to the spaces using a coordinate system. Figure [2](#_bookmark1) illustrates what coordinates correspond to which space. As you can see, every space is associated with a coordinate pair (*x, y*). However, the *x*-axis and *y*-axis are not orthogonal in the normal visualization.

Representation of Positions

The server sends you the position (game state) as a JSON object, indicating where each agent has its pegs. For example, the position shown in Figure [2](#_bookmark1) would be represented as

{"A": [[2, 1], [1, 0], [2, -1], [2, -3], [1, 1]],

"B": [[2, -2], [3, 0], [-2, 0], [-1, 1], [3, -1]],

"C": [[-1, 0], [-2, 3], [-4, 1], [3, 1], [-3, 1]]}

Your agent is always player *A*.

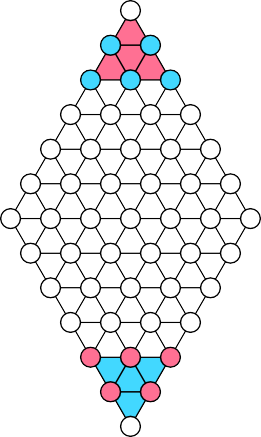
## Representation of Moves

Moves are represented as a JSON list of coordinate pairs, where the first coordinate pair indicates the peg you want to move and the last one indicates what space you want to move the peg to. If you have a hop chain, the list has to include all intermediate spaces, i.e. two consecutive list entries should be a simple hop apart from each other. For example, these would be valid moves for player *A* in Figure [2:](#_bookmark1)

* + - [[1, 0], [0, 1]]
    - [[1, 0], [3, -2], [1, -2], [3, -4]]

## Simplified variants

The server also has simplified variants of FAUhalma, using either the star-shaped board of FAUhalma or a rhombic board, which is basically a star-shaped board with some corners cut-off (Figure [3).](#_bookmark2) The variants have the same coordinate system and move rules as the standard variant of FAUhalma.

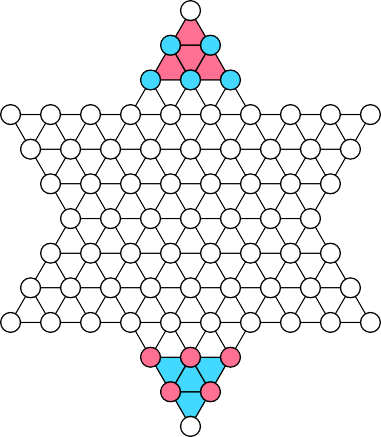


**B B**

**B B B**

**A A A**

**A A**



**B B**

**B B B**

**A A A**

**A A**

Figure 3: The starting positions of simplified FAUhalma variants.