Exercise 3 – Implementation works due to before christmas **[10 points]**

The rules for a round-based card game (in german probably known as „swimming“/Schwimmen, or 30 ½) for at least 2 players are as follows:

- initialization
  - a card deck (containing labels 7,8,9,10, kings, queens, knights, and aces in 4 suits – spades, hearts, diamonds, and clubs) is shuffled.
  - Each player except the starting one gets randomly three card.
  - The player starting the game has two options, as there are two potential desks for him. He may only have a look at one of those, and:
    - may decide to use that one for further playing, or
    - discard that one and use the other, for him unknown, cards for playing.
  - Depending on either action, the other cards from the remaining desk become visible to all players.
- The main game is based on the three visible cards. One player at a time may choose actions out of:
  - exchanging one card from her hand with one of the visible ones.
  - Doing nothing (pass over to the next player)
  - „closing“ the game
    - once a player closes the game, each other player may have once additional turn
- winning rules are as follows. Players try to maximize the sum of their hands.
  - Card values are
    - 7, 8, 9 each count 7, 8, or 9 points
    - 10, kings, queens and knights each account for 10 points
    - Pike counts 11 points.
  - Summing points is as follows:
    - for same-color cards, simply sum the numbers
    - having three equal cards of different suits (e.g., three '7's, three kings) always sum to 30 ½

In the „original“ game, there are more possible actions which may be ignored for simplicity.

Build groups of 2-3 people (larger groups need to justfy the required amount of people...). Implement the game by dividing the tasks as follows:

(1) implements the game field and interaction patterns (e.g., exchange card from hand to table, give information on which cards are visible on the table),
(2) and (3) implement independantly (!) one AI each.

Requirement: the AIs should play some games against each other. Which wins most?

Again in groups: Prepare a small oral speech (no slides, etc. needed) to present the game / your implementation to the others explaining why you implemented the requirements for the given component that way.