13 JANVIER 2023

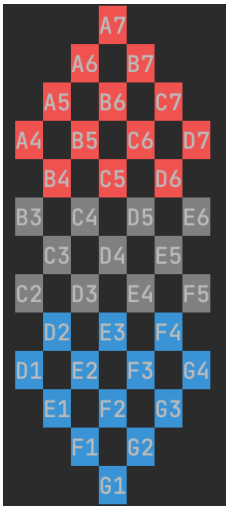
BESSON LUCAS
LADEL AMINE

SAE S1.02

STUCKWIN

PREAMBLE

To finish the first semester of our BUT in Computer Science, SAE 1.01-02 was proposed to us. The teacher in charge of this SAE asked us to create a board game called "StuckWin". Here is the game board and its rules:



-The game board, reproduced here, has 37 hexagonal spaces, or hexagons, spread out on a hexagonal board.

-The game includes 13 blue and 13 red identical tokens.

-One player holds the blue tokens, the other one holds the red tokens.

-Each player, in turn, moves a token to an empty space immediately in front of them. A token therefore has a maximum of three possible destination spaces. For example: at the beginning of the game, the blue token in E3 can only be moved to D3, D4, or E4.

- A token can not jump over one or more other tokens.

- If a player has no more possible moves when it is turn, he win the game.

INTRODUCTION :

SAE 1.01-02 was split into two parts. SAE 1.01 (which focused solely on the game's interface, rules, and proper functioning) and SAE 1.02 (on how to play with the creation of two artificial intelligences). In this report, we will only be interested in SAE 1.02.

In the first part, we will give brief details on the first AI, called the naive AI, and then continue with an explanation of the functioning of the AI, called the intelligent AI. Finally, we will demonstrate a graphical example in game of our intelligent AI.

I- THE AI'S

A- Naive AI game strategy.

As its name suggests, the naive AI plays without thinking. Indeed, it goes through the entire board to find the first token that it can move. Once the right token is found, it plays the first available space that it finds. Thus, it plays in a very similar way and its behavior becomes quite predictable.

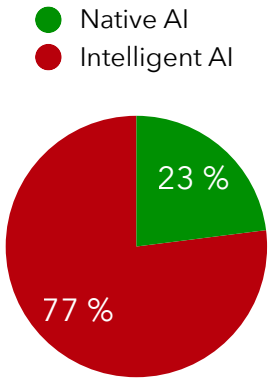
B- Intelligent AI game strategy.

Can we really call our program that plays in a naive way an 'AI'? In our opinion, the artificial intelligence of our game must meet the original definition of AI, particularly that it is necessary for it to adapt to each situation.

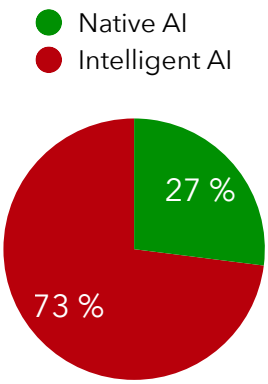
By following this simple rule, our first goal was to create an AI that generates thousands of games, which it analyzes in order to deduce the best strategy to adopt to win a game. Being constrained by the prohibition of using other files than "Stuckwin.java", we reworked the original idea to adapt it to the constraint.

It took a multitude of games played to finally try to understand a potential strategy. Thus, we were able to deduce a method that seems closest to the original idea, namely adapting to the opponent. It is necessary to understand for the sake of further explanations, that a player is in a position of strength when the opponent holds the most possibilities of playing on the entire board. Naturally, when we (humans) play, we try to imagine the continuation of the game by simulating in our intellect the possibilities of the opponent after our move. We tried to transcribe this psychological mechanism through our AI. Therefore, the AI scans the board looking for its pawns. As soon as it finds one, it checks if it can move it. If it is indeed able to make this move, it does so in 'its intellect' (in reality, a copy of the board state specific to it) the move in question. It then checks the number of possibilities that the opponent can play after simulating this move in memory. It applies this analysis technique to each pawn and stores in memory the best move to make. If it realizes that several moves generate the same number of possibilities for the opponent to play, it chooses a move at random to play. This "random" aspect of the AI favors the human at the beginning of the game since at the start, our number of possibilities is practically the same on the first moves. As a result, the game becomes increasingly difficult for the AI as there are fewer and fewer pawns with the same number of possibilities.

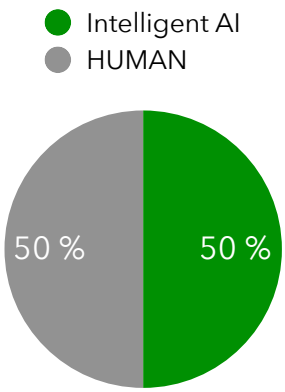
II- COMPARISON OF THE TWO AI'S :



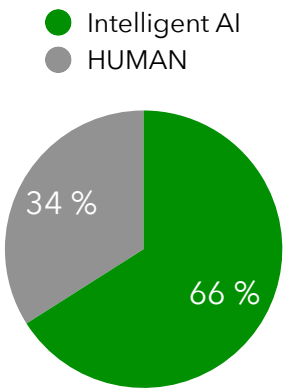
IA INTELLIGENTE
VS IA NATIVE
Score for 200 games played



IA INTELLIGENTE
VS IA NATIVE
Score for 20 000 games played



IA NATIVE
VS HUMAN
Score for 4 games played

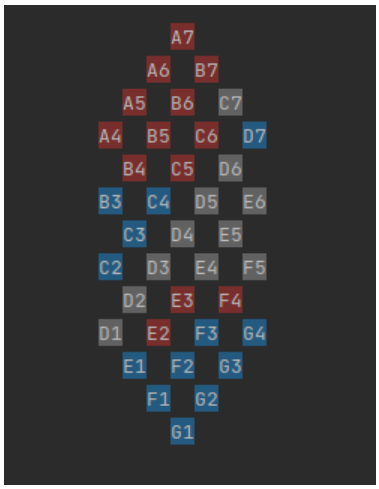


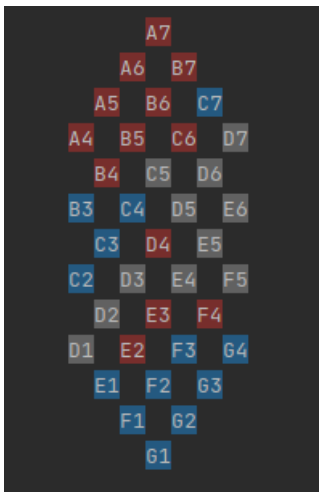
IA INTELLIGENTE
VS HUMAN
Score for 4 games played

Above you can find different graphs that represent the victory rate of each AI in percentage.

III- GRAPHIC EXAMPLES OF INTELLIGENT AI

In the example above, I am about to play the move D7->C7. The AI understands that this move offers more than one possibility to play. Thus, it will play the best move (here C5->D4), because it deduced that by playing this way, I will have to shift the entire diagonal to win, and thus my number of possibilities will become greater.





CONCLUSION

This project allowed us to develop a multitude of skills while remaining playful. Whether it's understanding the functioning of boards, their displays and their major uses in a game, this game also allowed us to get started with artificial intelligence.

We would like to thank the entire educational staff who participated in the organization of this project.