CODE IT

P.S.G College of Technology

INTRODUCTION:

The game is played between 2 players (one user and the other computer) on an 8X8 Squares on the board. Each player has 12 pegs. The game is to move the pegs (Circles). Red color is for user and green is for computer.

LOGIC AND DATASTRUCTURES:

The basic logic used to implement the Artificial Intelligence is to bring all the pegs together is by minimizing the distance between them (which is calculated by creating a minimum spanning tree with all the available pegs as the nodes) and at the same time trying to increase the distance among the opponent's pegs. This value is calculated as the sum of the edges on the Minimum Spanning Tree. But this sum is not sufficient. Consider the state given below.

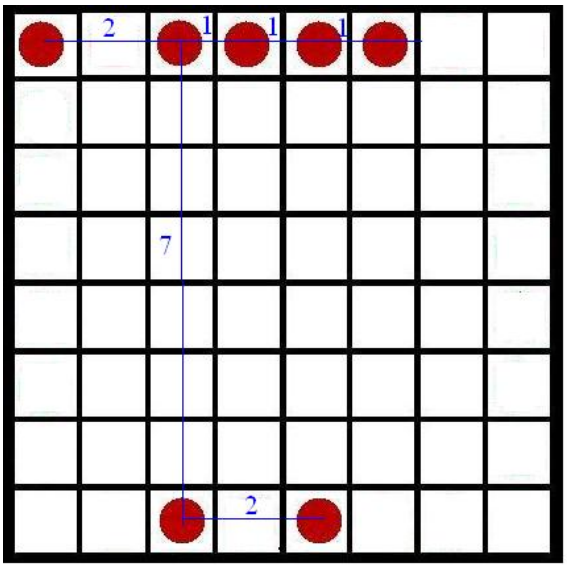


Fig (a)

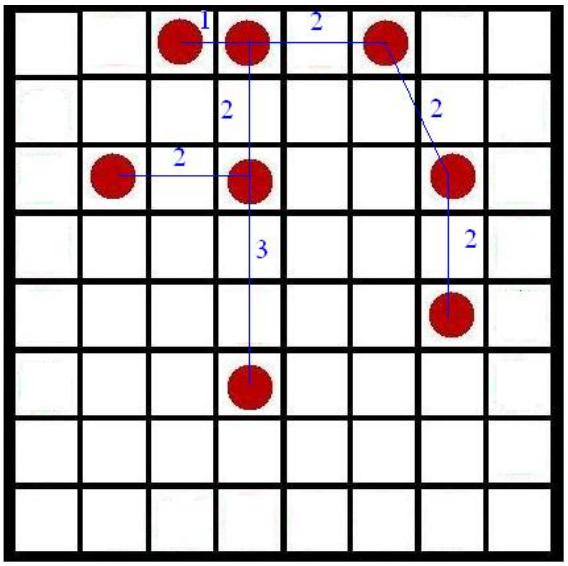


Fig (b)

Both Fig (a) and Fig(b) have the sum of the edges as 14, but Fig(b) is considered as a better position compared to Fig(a) as the pegs are more closer to each other and

moving a peg towards another would increase the chances of winning. Hence this distance factor is multiplied with another factor , which is the length of the largest edge of the Minimum Spanning Tree. Hence we define proximity factor as:

Proximity factor = Sum of length of edges of the MST × Length of edge with max distance

The proximity factor is then normalized to a range between -1 and 1, called the Win Factor. A Win Factor of value -1 indicates the user has won (computer has lost) and +1 indicates the computer has won. A game tree is created which identifies all the possible moves of all the pegs. The height of the game tree controls the number of steps the computer can think ahead. Increasing the height of the tree would indicate that the computer would think ‘h / 2’ moves ahead where h is the height of the tree. The Win Factor is calculated as

1. Calculate the Win Factor of the leaf nodes.
2. The Win Factor of the children is aggregated as the Win Factor of the parent.

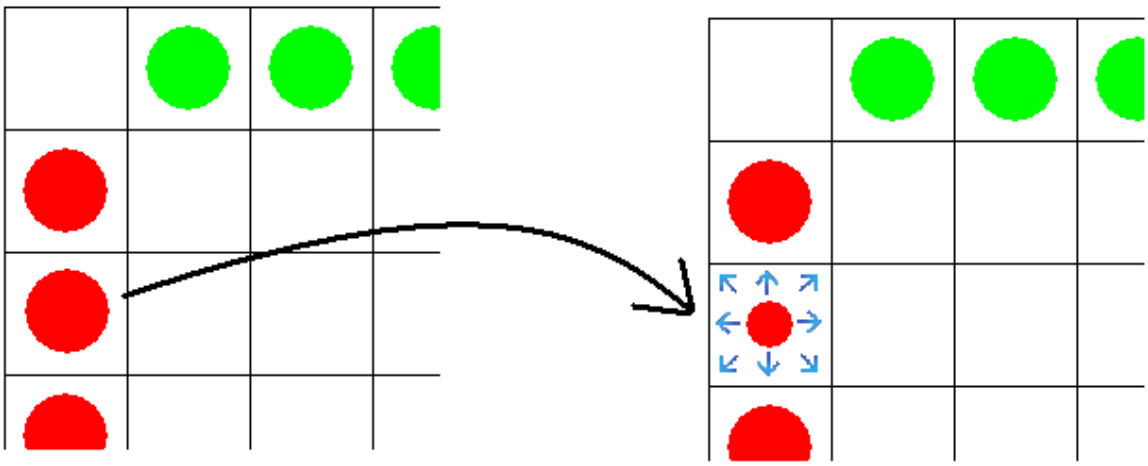
The move with the highest Win Factor is chosen from the Game Tree and hence the next best move is decided.

INTERFACE DESIGN:

The interface provided to the user is as follows

1. The Red circles represent the player’s pegs.
2. The Green circles represent the user’s pegs.
3. To move a peg, click on the peg
    - **a.** The size of the circle is reduced with 8 arrows around it, indicating the 8 directions in which the peg can move.
    - **b.** The user can click on any of the arrows to move the peg in that direction as shown in the figure.

4. Other details like invalid move, thinking (indicating the computer is evaluating the next move), etc. are shown below the board.



When a peg is selected

SOFTWARES REQUIRED FOR DEMO:

1. NetBeans 5.5+
2. JDK 1.5+