Assignment for CS F214: Logic In Computer Science

Tic-Tac-Toe Problem 1

Team:

Naigam Shah (2016A7PS0027G)

Arkil Patel (2016A7PS0665G)

Malhar Patel (2016A7PS0664G)

Abhimanyu Chaturvedi (2016A7PS0671G)

Introduction

We have made a program that is a Computer vs Human Player Tic-Tac-Toe Game. The computer is the First Player and will use the symbol “X” for his moves, while the Human Player is the Second Player and will use the symbol “O”.

By means of this program, we seek to demonstrate that by adding just one new rule to the game, while giving the Computer Player the first move, the computer has a winning strategy, and by playing optimally, it shall utilise this winning strategy, and always be guaranteed a win.

What does Playing Optimally mean? And why is it necessary?

Playing optimally means that the computer will always make the best possible move in any situation that results in it utilising the effective strategy to win the game.

Statement: It is absolutely necessary that the computer play optimally to make use of the winning strategy.

Proof: Assume that Computer is NOT playing optimally. This means that it is making its move at random.

Given a situation as follows -

X	O	O
1	O	3
X	2	X

Situation - A

Since the computer is making its move at random, it won't be able to recognise the win opportunity that it has (by putting X in position 1 or 2). It might just as well put X in 3 and lose the game despite having a winning strategy.

Hence, in order to use the winning strategy leading to a win, the computer must play optimally.

What exactly is the Computer Player doing that counts as playing Optimally?

The centre space in a Tic-Tac-Toe game is the most strategic space. It has interactions with all other places and is also essential in the win through two diagonals and one row and one column.

Since the computer is the first player, it has the opportunity to occupy any space in the first chance. So, it will occupy the centre space.

Now, based on the inputs of the human player, we have developed our algorithm to input the computer's move according to the winning strategy.

The algorithm will identify the situation each time after the human player's move, and decide the best possible move for the Computer.

The New Rule

We have introduced a new rule which will always guarantee a win to the first player (i.e. Computer Player). The game proceeds as usual until the final space is left to be filled. After the 8 normal moves (4 from each player) are done, and the game isn't over yet, now there is a condition to the next move/s. The players can add their input in the one empty space left only after removing/erasing their previous input in an adjacent space (adjacent does not include diagonally adjacent). So practically, one space will always remain empty and players fill it according to the above rule until there is a winner.

For example, let's say after 8 moves, the situation is as follows: (Suppose X is the first player)

	O	X
X	O	O
O	X	X

Situation - B

Now it is X's chance. It has only one possible option, i.e. putting X in the empty space and erasing the X adjacent.

X	O	X
	O	O
O	X	X

Situation - C

Now, it is O's chance. It has two possible moves. After it has put the O in the empty space, it can either erase the O in the 3rd row (1st column) or it can erase the O in the 2nd row (middle block). Clearly, erasing the O in the 3rd row will result in its win.

X	O	X
~~O~~	~~O~~	~~O~~
	X	X

Situation – D

Note that in the above case, the second move player (i.e. O) won the game. This means that X, despite having the advantage of the first move, failed to play optimally.

In our program, the Computer player has the first move and since it plays optimally, it will always win following this new rule. This is demonstrated in the program.

How and why did we think of this rule?

During our various encounters with the Tic-Tac-Toe game, we observed that if the first player played optimally, the game will result in either his win or a draw, but never a loss. However, we were tasked with finding a winning strategy. So, we carefully separated all the situations which resulted in a draw and observed them.

After this it was all trial and error. We tried imposing various rules, and after various failures, we finally decided upon the aforementioned rule, which guarantees a win to the computer player if played optimally.