**Description:**

We have coded the tic-tac-toe game. The game has two modes. They are,

1. Player 1 vs Player 2
2. Player vs CPU

The game runs in the command window. At each level in the game, the players receive appropriate instructions on how to proceed. The game can be won if a player can successfully have their symbol in 3 consecutive boxes in the board, either vertically, horizontally or diagonally.

The 1st mode is fairly straightforward. Player 1 goes first. The game continues till one player wins or the game ends in a tie. Each player make their moves alternately.

In order to increase the complexity of the code and the game itself, for the 2nd mode, we have used the AI technique, Minimax algorithm with Alpha-Beta pruning. This ensures that when playing against the computer, it never loses. This algorithm is implemented to find the best move in every possible situation and the computer will always choose this move. A minimax function is written in the code which outputs this move.

**Minimax algorithm with Alpha-Beta pruning:**

Alpha-Beta pruning is an optimization technique for minimax algorithm. It reduces the computation time by a huge factor. This allows us to search much faster and even go into deeper levels in the game tree. It cuts off branches in the game tree which need not be searched because there already exists a better move available. It is called Alpha-Beta pruning because it passes 2 extra parameters in the minimax function, namely alpha and beta.

* **Alpha** is the best value that the **maximizer** currently can guarantee at that level or above.
* **Beta** is the best value that the **minimizer** currently can guarantee at that level or above.