##### Assignment - Report

##### Artificial Intelligence – UE17CS325

**TOPIC** : TICTACTOE GAME IMPLEMENTATION USING ALPHA BETA PRUNING .

**TEAM MEMBERS:**

JNANESH D : PES1201701822

RAKESH DEVANI : PES1201701602

ABHILASH H: PES1201701644

**ABSTRACT :**

User vs Computer classical game ‘Tic tac toe’ Is implemented using alpha beta pruning in python .This problem consists of initializing the game board object, user turn playing the game, computer turn and evaluating the board . We implemented the game by initializing the game board 3x3.

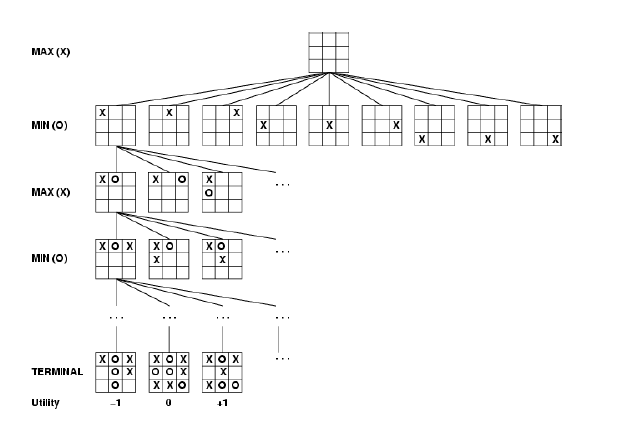
User has to enter the index of the his move position. The algorithm generates the optimal move for the computer.

The alpha beta pruning is a recursive algorithm which searches the best move for a computer by searching the nodes by dfs and backing up the alpha beta values upto the root . This algorithm reduces the branching factor of the search space compared to the Minimax algorithm . User is MAX player and computer is the MIN player .

# Introduction

Tictactoe is a deterministic game with perfect information.Two agents Human and computer whose actions alternate. Utility values for each agent are opposite to each other. Environment in this game is fully observable .

Game tree for tictactoe



Here MAX player is human and MIN player is Computer.

**Formal definition of problem:**

**Initial state:** Empty state of board

\_ | \_ | \_

\_ | \_ | \_

| \_ |

**Player\_Move:** Defines human player has to make valid move

**Action:** Returns a move for the computer

**Terminal-Test(s):** Is the game finished? True if finished, false otherwise.

**Utility function(s,p):** Gives numerical value of terminal state s for player p win (+1), lose (-1), and draw (0)

**Approach:**

Coding is done in python. We implemented a board class.An evaluation function in that class for result evaluation and isMoveLeft function to get whether any moves left in the board object.

A game class which contains modules AlphaBetaPruning and NextMoveAlphaBeta. NextMoveAlphaBeta makes a call to AlphBetaPruning passing the alpha beta and board state .It returns the board object with the computer’s move updated

### Alpha beta searching :

After user’s move the algorithm for next-move-by-alpha-beta is applied for the computer’s move to calculate the best move . Alpha beta pruning is an optimization of the minimax algorithm.

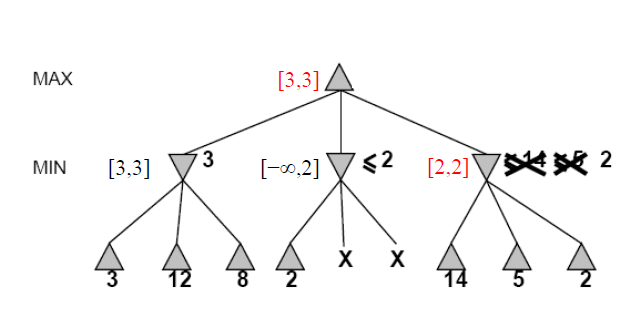
Strategy is the algorithm assigns the empty spaces of the board as computer’s move and evaluates the state of board by alpha beta pruning

**Prune whenever α ≥ β.**

Prune below a Max node whose alpha value becomes greater than or equal to the beta value of its ancestors. **Max nodes update alpha** based on children’s returned values.

Prune below a Min node whose beta

value becomes less than or equal to the alpha value of its ancestors. **Min nodes update beta** based on children’s returned values



Above pic is an example of pruning strategy.

α = highest-value choice found at any choice point of path for MAX

(initially, α = −infinity)

β = lowest-value choice found at any choice point of path for MIN

(initially, β = +infinity)

**Pass current values of** α **and** β **down to child nodes during search.**

•**Update values of** α **and** β **during search:**

–MAX updates α at MAX nodes

–MIN updates β at MIN nodes

• **Prune remaining branches at a node when** α **≥** β

**Test Cases**

We tested the results based on the optimality of the move generated by the algorithm and time complexity of the finding that move for the computer’s turn each time.

**Conclusion**:

The Alpha-Beta implementation of the TicTacToe is a procedure which can prune large parts of the search tree of minimax . Alpha-beta is guaranteed to compute the same value for the root node as computed by minimax, with less or equal computation

\* Worst case: no pruning, examining b^d leaf nodes, where each node has b children and a d-ply search is performed

\* Best case: examine only (2b)^ d/2 leaf nodes. – Result is you can search twice as deep as minimax!

\* Best case is when each player’s best move is the first alternative generated