LAB 07

Alpha Beta Pruning (Tic Tac Toe)

Exercise

1. Implement the AlphaBeta Pruning in Tic Tac Toe Game

Code

import math

board = [" " for \_ in *range*(9)]

*def* **print\_board**():

    for i in *range*(0,9,3):

*print*("| "+" | ".*join*(board[i:i+3])+" |")

*def* **check\_winner**(player):

    win\_patterns = [

        [0, 1, 2], [3, 4, 5], [6, 7, 8],

        [0, 3, 6], [1, 4, 7], [2, 5, 8],

        [0, 4, 8], [2, 4, 6]

    ]

    for pattern in win\_patterns:

        if *all*(board[i]==playerforiinpattern):

            return True

    return False

*def* **minimax\_alpha\_beta**(is\_ai, alpha, beta):

    if *check\_winner*("O"):

        return 1

    if *check\_winner*("X"):

        return -1

    if " " not in board:

        return 0

    if is\_ai:

        best\_score = -math.inf

        for i in *range*(9):

            if board[i] == " ":

                board[i] = "O"

                score = *minimax\_alpha\_beta*(*False*,alpha,beta)

                board[i] = " "

                best\_score = *max*(score,best\_score)

                alpha = *max*(alpha,score)

                if beta <= alpha:

                    break

        return best\_score

    else:

        best\_score = math.inf

        for i in *range*(9):

            if board[i] == " ":

                board[i] = "X"

                score = *minimax\_alpha\_beta*(*True*,alpha,beta)

                board[i] = " "

                best\_score = *min*(score,best\_score)

                beta = *min*(beta,score)

                if beta <= alpha:

                    break

        return best\_score

*def* **best\_move**():

    best\_score = -math.inf

    move = 0

    for i in *range*(9):

        if board[i] == " ":

            board[i] = "O"

            score = *minimax\_alpha\_beta*(*False*,-*math*.inf, *math*.inf)

            board[i] = " "

            if score > best\_score:

                best\_score = score

                move = i

    board[move] = "O"

*def* **play\_game**():

*print\_board*()

    while " " in board:

        player\_move = *int*(*input*("Enter your move (0-8): "))

        if board[player\_move] == " ":

            board[player\_move] = "X"

            if *check\_winner*("X"):

*print\_board*()

*print*("You win!")

                return

*best\_move*()

*print\_board*()

            if *check\_winner*("O"):

*print*("AI wins!")

                return

        else:

*print*("Cell taken, try again.")

*print*("It's a draw!")

*play\_game*()

Output



