IMPLEMENTATION OF MINIMAX algorithm

R.K.Viswesh

241801319

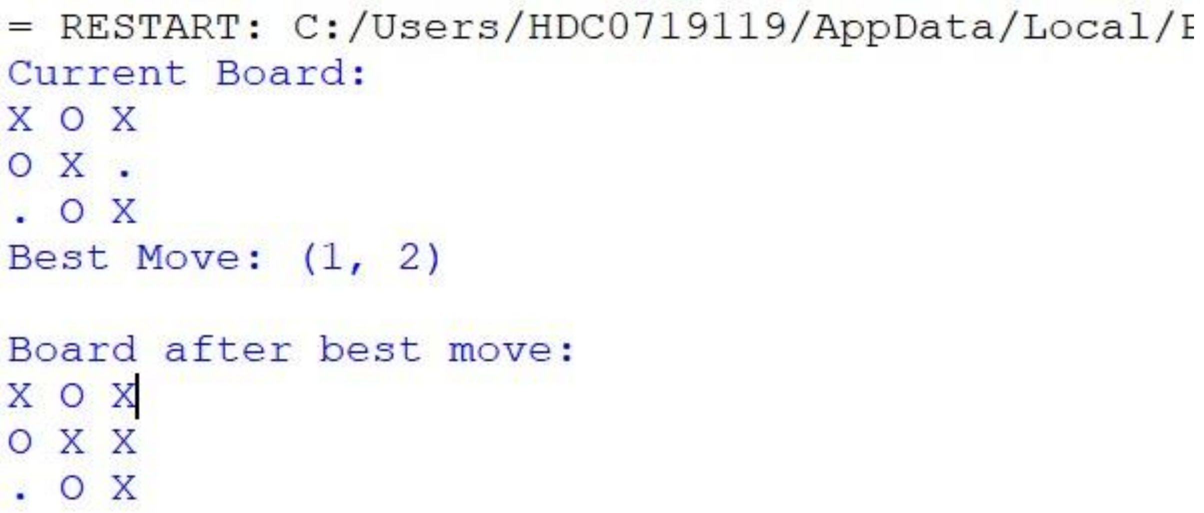
AIDS-FD

PROGRAM:

```
# Constants for players
X = 1
O = -1
EMPTY = 0

# Evaluate the board def evaluate(board):     for row in range(3):
if board[row][0] == board[row][1] == board[row][2] != EMPTY:
            return board[row][0]     for col in range(3):        if
board[0][col] == board[1][col] == board[2][col] != EMPTY:
            return board[0][col]     if board[0][0] ==
board[1][1] == board[2][2] != EMPTY:
        return board[0][0]     if board[0][2] == board[1][1]
== board[2][0] != EMPTY:
        return board[0][2]
return 0

# Check if moves are left def
hasMovesLeft(board):     for row in
```

```
range(3):        for col in range(3):
if board[row][col] == EMPTY:
                return True
return False


# Minimax function def
minimax(board, isMax):
score = evaluate(board)     if
score == X:         return score
if score == O:         return
score     if not
hasMovesLeft(board):
return 0     if isMax:
        best = -float('inf')
for row in range(3):
for col in range(3):
                if board[row][col] == EMPTY:
                    board[row][col] = X
                    best = max(best, minimax(board, not isMax))
board[row][col] = EMPTY         return best
    else:
        best = float('inf')         for row in
range(3):            for col in range(3):
if board[row][col] == EMPTY:
                    board[row][col] = O
                    best = min(best, minimax(board, not isMax))
board[row][col] = EMPTY         return best


# Find the best move for X def
findBestMove(board):
   bestVal = -float('inf')
bestMove = (-1, -1)     for
row in range(3):         for
col in range(3):
          if board[row][col] == EMPTY:
             board[row][col] = X
moveVal = minimax(board, False)
board[row][col] = EMPTY                 if
moveVal > bestVal:                    bestMove
= (row, col)                    bestVal = moveVal
return bestMove


# Print the board
def printBoard(board):
for row in board:
      print(" ".join(["X" if x == X else "O" if x == O else "." for x in row]))


# Example game board
= [
   [X, O, X],
   [O, X, EMPTY],
    [EMPTY, O, X]
]

print("Current Board:")
printBoard(board)

move = findBestMove(board)
print(f"Best Move: {move}")

board[move[0]][move[1]] = X

print("\nBoard after best move:")
printBoard(board)
```

OUTPUT:



```
= RESTART: C:/Users/HDC0719119/AppData/Local/
Current Board:
X O X
O X .
. O X
Best Move: (1, 2)

Board after best move:
X O X
O X X
. O X
```