Implementing a Game of Tic-Tac-Toe in Python Using a MiniMax Algorithm as the Computer

Team Falcon Punch, IS201 – Fundamentals of Computing, School of Technology & Computing, City University of Seattle

[halleric@cityu.edu](mailto:halleric@cityu.edu), [willcoxsonwillie@cityuniversity.edu](mailto:willcoxsonwillie@cityuniversity.edu), [perduedonald@cityuniversity.edu](mailto:perduedonald@cityuniversity.edu)

**Abstract**

For Tic-Tac-Toe, the following pseudocode will be a guideline for establishing the rest of the code.  
  
def clear\_screen to clear the previous game and ready the gameboard for another game

def display\_board which will call the clear\_screen function and then build the playing board

def check\_winner lists all the winning combinations in list form based on the current game board and the numerical position and then uses an if statement nested within a for loop to determine if a winning combo is found. They include:

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

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

Diagonals: [0, 4, 8], [2, 4, 6]

def valid\_input:

try:

move = player input

if else loop to check for valid move

except valueError that prints for numbers chosen outside 1 through 9

def minimax algorithm for computer move:

will need to determine what the best score could possibly be depending on the moves available

this best score will be returned and then used in a find\_best\_move function

def find\_best\_move:

initialize best\_score and best\_move

for i in range(9):

if board[i] == “”

board = X or O

score = minimax function

board[i] = “”

if score > best\_score:

best\_score = score

best\_move = i

return best\_move

def log\_game\_result(result):

with open log result using write(result)

def play\_tic\_tac\_toe:

set up board

define player and computer

Ask the user if they want to play first and use a while loop to get input and then set variables for the player and computer. Use another loop to determine if the player wants X or O

another while loop to display the board

if else for performing moves for both player and computer

set board move for current player

perform if statements for winner checks and if the board is full or not

if statement for the \_\_main\_\_ with a while boolean with true clearing screen and playing tic-tac-toe. Bottom of the while loop will have an if statement for playing again and break if not yes

**Keywords:** lists, loops, minimax, file logging, and breaks

**9.** **REFERENCES**

GeeksforGeeks. (2023). Python def Keyword. *GeeksforGeeks*. https://www.geeksforgeeks.org/python-def-keyword/

Klinger-Meyers, B. (2022). How to always win at Tic Tac Toe | Best Tic tac toe Strategy. *Twinfinite*. <https://twinfinite.net/guides/how-to-always-win-at-tic-tac-toe-best-tic-tac-toe-strategy/>

Matthes, E. (n.d.). *Python Crash Course, 2nd Edition*. O’Reilly Online Learning. <https://learning.oreilly.com/library/view/python-crash-course/9781492071266/xhtml/ch04.xhtml>

Python, R. (2023). Minimax in Python: Learn how to lose the game of NIM. *realpython.com*. <https://realpython.com/python-minimax-nim/>