Date ___/___/___
Page _____

TICTACTOE

```
board = {
    1: ' ', 2: ' ', 3: ' ',
    4: ' ', 5: ' ', 6: ' ',
    7: ' ', 8: ' ', 9: ' ',
}

def printBoard(board):
    print(board[1] + '|' + board[2] + '|' + board[3])
    print('-+-+-')
    print(board[4] + '|' + board[5] + '|' + board[6])
    print('-+-+-')
    print(board[7] + '|' + board[8] + '|' + board[9])
    print('\n')

def spaceFree(pos):
    return board[pos] == ' '

def checkWin():
    winning_conditions = [
        (1,2,3), (4,5,6), (7,8,9),
        (1,4,7), (2,5,8), (3,6,9),
        (1,5,9), (3,5,7)
    ]
    for a,b,c in winning_conditions:
        if board[a] == board[b] == board[c] and
        board[a] != ' ':
            return True
    return False
```

```
def checkMoveForWin(move):
    winning_conditions = [
        (1,2,3),(4,5,6),(7,8,9),
        (1,4,9),(2,5,8),(3,6,9),
        (1,5,9),(3,5,7)
    ]
    for a,b,c in winning_conditions:
        if board[a] == board[b] == move and
        board[a] != ' ':
            return True
    return False

def checkDraw():
    return all(board[key] != ' ' for key in board)

def insertLetter(letter,position):
    if spaceFree(position):
        board[position] = letter
        printBoard(board)
        if checkDraw():
            print('Draw!')
        elif checkWin():
            if letter == 'X':
                print('Bot wins!')
            else:
                print('You win!')
        return
    else:
        print('position taken, please pick a different
              position.')
        position = int(input('Enter new position:'))
        insertLetter(letter,position)
player = 'O'
bot = 'X'
```

```
def playerMove():
    position = int(input('Enter position for O:'))
    insertLetter(player,position)

def compMove():
    bestScore = -1000
    bestMove = 0
    for key in board.keys():
        if board[key] == ' ':
            board[key] = bot
            score = minimax(board,False)
            board[key] = ' '
            if score > bestScore:
                bestScore = score
                bestMove = key
    insertLetter(bot,bestMove)

def minimax(board,isMaximising):
    if checkMoveForWin(bot):
        return 1
    elif checkMoveForWin(player):
        return -1
    elif checkDraw():
        return 0

    if isMaximising:
        bestScore = -1000
        for key in board.keys():
            if board[key] == ' ':
                board[key] = bot
                score = minimax(board,False)
                board[key] = ' '
                bestScore = max(score,bestScore)
```

Date __/__/__
Page _____

```
        return bestScore
    else:
        bestScore = 1000
        for key in board.keys():
            if board[key] == ' ':
                board[key] = player
                score = minimax(board, True)
                board[key] = ' '
                bestScore = min(score, bestScore)
        return bestScore

# Game loop
while not checkWin() and not checkDraw():
    compMove()
    if not checkWin() and not checkDraw():
        playerMove()
```