24/09

Tic Tac Toe

Algorithm:

Step 1: Create an empty tic tac toe board as a 2D matrix

b = [[' ', ' ', ' '], [' ', ' ', ' '], [' ', ' ', ' ']]

Step 2: Let the user play the first move,
move = int(input("Enter the no of square"))

Step 3: ~~Mark the~~ Create a set of all the win conditions

Step 4: Mark the user move in the board as 'O'

Step 5: Check if the user has won. If so, display user wins and go to step 10

~~Step 5: def check almost(a):~~

~~Step 5~~: Check if the board is full. If it display tie. Go to step 10

Step 6: Check if the computer can win the game in this move. Check if any row, column or diagonals have two 'X' and one space ' '. If there is, make that move and display computer wins
Go to step 10

Step 7: Check if the player can win the game in the next move. Check if any row, column or diagonals have two 'O's and one space ' '. If there is, mark that space as 'X'. Go to step 2

Step 8: If the middle square is empty, mark the move as 'X'. Go to step 2

Step 9: Out of the remaining spaces, pick a random

square and mark it as 'X'. Go to step 9

Step 10: Game over

Code:

```
import random

b = [[' ',' ',' '],[' ',' ',' '],[' ',' ',' ']]

def isWin(a):
    for i in range(3):
        if all(b[i][j] == a for j in range(3))
            or all(b[j][i] == a for j in range(3)):
                return True
        if all(b[i][i] == a for i in range(3))
            or all(b[i][2-i] == a for i in range(3)):
                return True
    return False

def isFull():
    return all(cell != ' ' for row in b for cell in r

def canWin(a):
    for i in range(3):
        if b[i].count(a) == 2 and b[i].count(' ') ==
            return (i, b[i].index(' '))
    for i in range(3):
        l = [b[j][i] for j in range(3)]
        if l.count(a) == 2 and l.count(' ') == 1:
            return (l.index(' '), i)
    diag1 = [b[i][i] for i in range(3)]
```

```
    if diag1.count(a) == 2 and diag1.count(' ') == 1:
        return (diag1.index(' '), diag1.index(' '))
    diag2 = [b[i][2-i] for i in range(3)]
    if diag2.count(a) == 2 and diag2.count(' ') == 1:
        return (diag2.index(' '), 2 - diag2.index(' '))
    return (-1, -1)

def display():
    print(b[0])
    print(b[1])
    print(b[2])

while True:
    display()
    x = int(input("Enter row: "))
    y = int(input("Enter column: "))
    if b[x][y] != ' ':
        print("Invalid move!")
        continue
    b[x][y] = 'O'
    if isfull():
        display()
        print("Tie!")
        break
    if isWin('O'):
        display()
        print("You win!")
        break
    (p, q) = canWin('X')
    if (p, q) != (-1, -1):
        b[p][q] = 'X'
    display()
    print("Computer Wins!")
```

```
        break
    (p,q) = canWin('O')
    if (p,q) != (-1,-1):
        b[p][q] = 'X'
        continue
if b[1][1] == ' ':
    b[1][1] = 'X'
else:
    l = [(i,j) for i in range(3) for j in range
        if b[i][j] == ' ']
    ran = random.choice(l)
    b[ran[0]][ran[1]] = 'X'
```