Date 24/09/24
Page 1

LAB 1

TIC TAC TOE! (user v/s computer)

Algorithm.

Step 1 → create a 3×3 matrix which will give 9 empty boxes.

Step 2 → Assign X or O to the user depending on the symbol he chooses And then assign the other to the computer. Assign – to represent empty spaces.

Step 3 → If the user starts the next turn will be given to the computer. this will happen till one of them win or till all the null spaces are filled which results in a tie.

Step 4 → If Now if the user starts, the computer should place O in such a way that it should not let the user make a row, a column or a diagonal with X's. (basically should not let the user win).

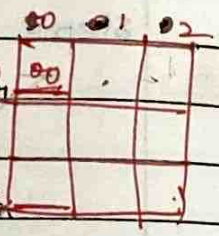
Step 5 → If the user or the computer creats a row or a column or a

Date / /
Page

diagonal with three respective symbols, will be the winner of that particular round in the tic-tak-toe game.

or

If all the moves are done and there are no empty space left and non of them have made a row or a column or a diagonal then it will be considered as a tie.




```
{ if(a[i][0] == a[i][1] == a[i][2] == player
  if[0][i] == a[1][i] == a[2][i] == player
winning condition
  if[0][0] == a[1][1] == a[2][2] == player
  if[0][2] == a[1][1] == a[2][0] == player
```

[if a[i][j] == 'x' if a[i][j] == player]

Date __/__/__
Page _____

code :-

```
import random
def print_board(board):
    for row in board:
        print("|".join(row))
        print("-" * 9)

def check_winner(board):
    for i in range(3):
        if board[i][0] == board[i][1] == board[i][2] != " ":
            return board[i][0]
        if board[0][i] == board[1][i] == board[2][i] != " ":
            return board[0][i]
    if board[0][0] == board[1][1] == board[2][2] != " ":
        return board[0][0]
    if board[0][2] == board[1][1] == board[2][0] != " ":
        return board[0][2]
    return None

def is_board_full(board):
    return all(cell != " " for row in board for cell in row)

def ai_move(board):
    for i in range(3):
        for j in range(3):
            if board[i][j] == " ":
                board[i][j] = "O"
                if check_winner(board) == "O":
                    return
                board[i][j] = " "
```

Date __/__/__
Page _____

```
for i in range(3):
    for j in range(3):
        if board[i][j] == " ":
            board[i][j] = "X"
            if check_winner(board) == "X":
                board[i][j] = "O"
                return
            board[i][j] = " "
if board[1][1] == " ":
    board[1][1] = "O"
    return

corners = [(0,0), (0,2), (2,0), (2,2)]
random.shuffle(corners)
for corner in corners:
    if board[corner[0]][corner[1]] == " ":
        board[corner[0]][corner[1]] = "O"
        return

sides = [(0,1), (1,0), (1,2), (2,1)]
random.shuffle(sides)
for side in sides:
    if board[side[0]][side[1]] == " ":
        board[side[0]][side[1]] = "O"
        return

def play_game():
    board = [[" " for _ in range(3)] for _ in range(3)]
    print("Welcome to tic tac toe!")
    print_board(board)
```

Date __/__/__
Page _____

```
while True :
    while True :
        try
            row = int (input (" Enter row (1-3) :")) -1
            col = int (input (" Enter column (1-3) : ")) -1
            if board [row] [col ] == " "
                board [row] [col] = "x"
                break
        else :
            print (" cell already taken, choose
                    another.")
    except (value.Error, IndexError):
        print (" Invalid input. Please enter numbers
                between 1 and 3. ")
    print_board (board)
    if check_winner (board) == "x" :
        print (" You win! ")
        break
    if is_board_winfull (board):
        print (" It's a draw! ")
        break.

    print ("AI's turn ... ")
    ai_move (board)
    print_board (board)

    if check_winner (board ) == "0":
        print ("AI wins] ")
        break
    if is_board_full (board ):
        print ("It's a draw ! ")
```

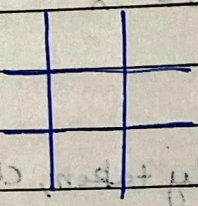
Date
Page

break

if __name__ == "__main__"

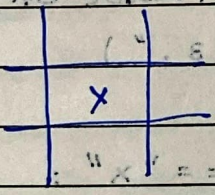
play_game()

Output



Enter row (1-3): 2

Enter row (1-3): 2

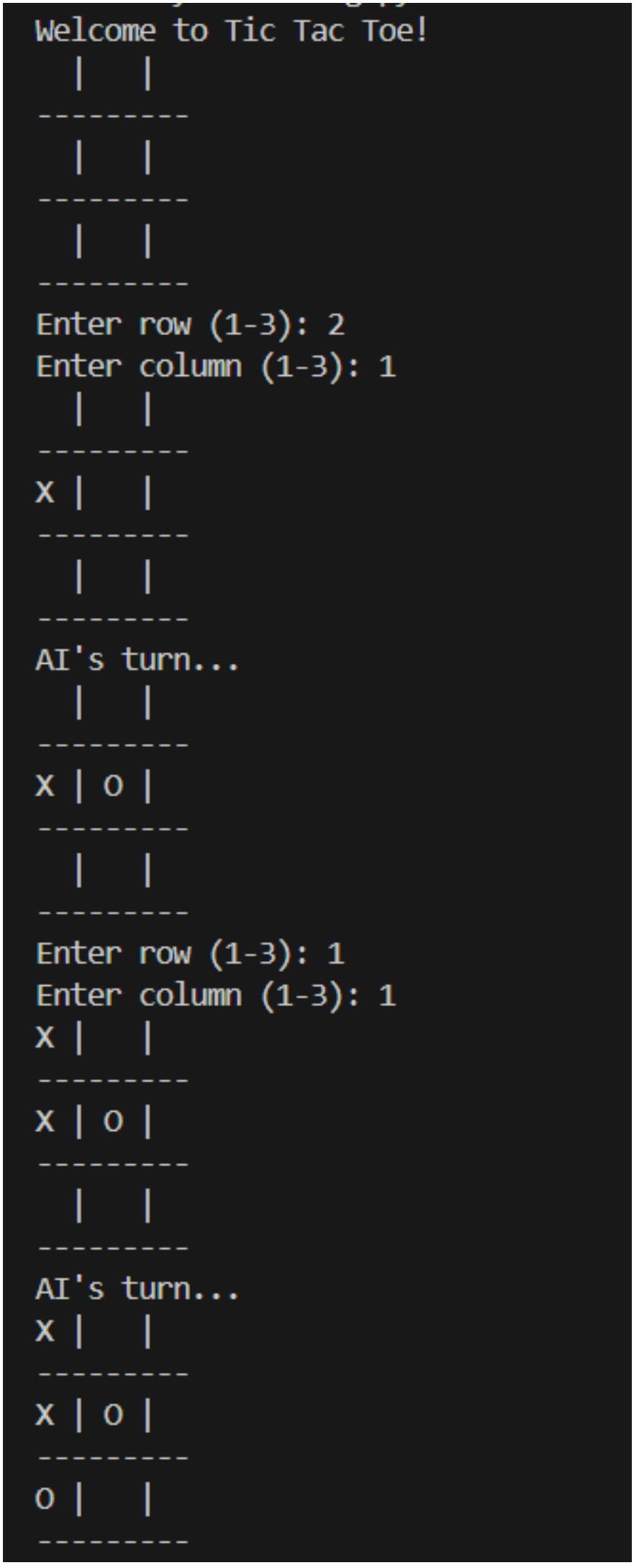


last matrix

O	X	O
O	X	
O		X

AI wins!

from 24/9/24


Welcome to Tic Tac Toe!
| |

| |

| |

Enter row (1-3): 2
Enter column (1-3): 1
| |

X | |

| |

AI's turn...
| |

X | O |

| |

Enter row (1-3): 1
Enter column (1-3): 1
X | |

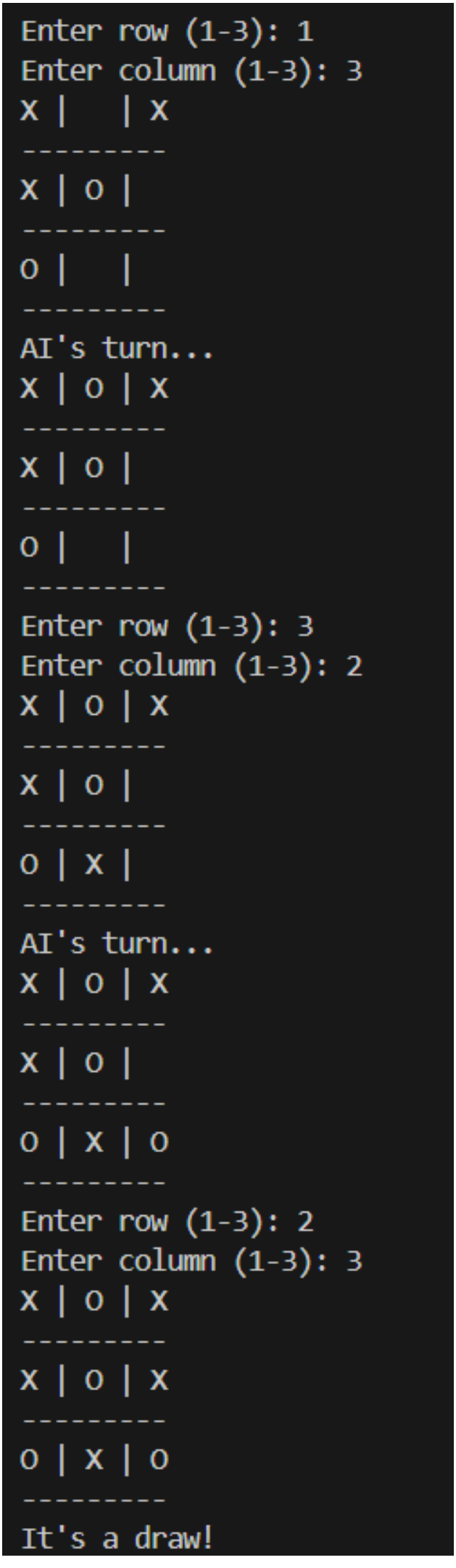
X | O |

| |

AI's turn...
X | |

X | O |

O | |


Enter row (1-3): 1
Enter column (1-3): 3
X | | X

X | O |

O | |

AI's turn...
X | O | X

X | O |

O | |

Enter row (1-3): 3
Enter column (1-3): 2
X | O | X

X | O |

O | X |

AI's turn...
X | O | X

X | O |

O | X | O

Enter row (1-3): 2
Enter column (1-3): 3
X | O | X

X | O | X

O | X | O

It's a draw!

```
---------
Enter row (1-3): 1
Enter column (1-3): 2
X | X |
---------
  | O |
---------
  |   |
---------
AI's turn...
X | X | O
---------
  | O |
---------
  |   |
---------
Enter row (1-3): 1
Enter column (1-3): 2
Cell already taken, choose another.
Enter row (1-3): 3
Enter column (1-3): 1
X | X | O
---------
  | O |
---------
X |   |
---------
AI's turn...
X | X | O
---------
O | O |
---------
X |   |
---------
Enter row (1-3): 3
Enter column (1-3): 2
X | X | O
---------
O | O |
---------
X | X |
---------
AI's turn...
X | X | O
---------
O | O | O
---------
X | X |
---------
AI wins!
```