DATE: PAGE:

1/10/24

Lab-1
Tic-toe Game

Algorithm
Step1: import numpy and random libraries
Step2: Create a function create board and creat an empty array as a board.
Step3: check for empty places in the board.

Pseudocode

Algorithm:
set t_board = [" ", " ", " ", " ", " ", " ", " ", " ", " "]

```
print_board()
  print(board[0] + "|" board[1] + "|" + board[2])
  print(board[3] + "|" + board[4] + "|" + board[5])
  print(board[6] + "|" + board[7] + "|" + board[8])
take_turn(player)
  position = input("chose a position from 1-9")
  while position not in ["1", "2" "3" "4" "5" "6" "7" "8" "9"
  pos: position = input("enter 1-9")
       position = int(position)-1
  while board[position] != " ":
       position = int(input("position already taken"))
       board[position] = player
       print(board) board()
  check_game_over()
  if board[0] == board[1] == board[2] != " " or
     board[3] == board[4] == board[5] != " " or
```

DATE:

PAGE:

```
board[2] == board[4] == board[6]
        return win
    elif " " in board:
        return "not in board"
    else:
        return "tie"
    return "play"
play_game()
print_board()
current_player = "X"
game_over = False
while not game_over:
    take_turn(current_player)
    game_result = check_game()
    if game_result == "win":
        print(current_player + "wins")
        game_over = True
    if game_result == "tie":
        print("It's a tie")
        game_over = True
    else:
        if current_player == "X":
            current_player = "O"
        else:
            current_player = "X"
play_game()
```

Output

```
_|_|_
_|_|_
 |_|_
X's turn
```

DATE: PAGE:

choose a position from 1-9 [1]

X | _ | _

_ | _ | _

_ | _ | _

O's turn

choose position 1-9 [2]

X | O | _

_ | _ | _

_ | _ | _

X's turn

choose pos 1-9 [4]

X | O | _

X | _ | _

_ | _ | _

O's turn [5]

X | O | _

X | O | _

_ | _ | _

X's turn

X | O | _

X | O | _

X | _ | _

"X" wins