Name: **Maaz Ihsan**

Roll No: **F24-620**

Program: **BSSE**

Semester/Section: 2nd

Subject: **OOP**

Session: **Spring 2025**

Submitted To: **MR. JAMAL ABDUL AHAD**

Department of Computer Science

Abbottabad University of Science & Technology

Project name:

- ## <u>TIC TAC TOE GAME</u>

<u>CODING</u>

```python
import tkinter as tk
from tkinter import messagebox

def check_winner():
    for combo in [[0, 1, 2], [3, 4, 5], [6, 7, 8],
                  [0, 3, 6], [1, 4, 7], [2, 5, 8],
                  [0, 4, 8], [2, 4, 6]]:
        if (buttons[combo[0]]["text"] ==
buttons[combo[1]]["text"] ==
buttons[combo[2]]["text"] != ""):
            for i in combo:
                buttons[i].config(bg='green')
            messagebox.showinfo("Game Over",
f"{buttons[combo[0]]['text']} wins!")
            root.quit()

def button_click(index):
    if buttons[index]["text"] == "":
        buttons[index].config(text=current_player)
        check_winner()
        toggle_player()

def toggle_player():
    global current_player
    current_player = "O" if current_player ==
"X" else "X"
```

```python
        label.config(text=f"Player:
{current_player}'s turn")

root = tk.Tk()
root.title("Tic Tac Toe")

buttons = [tk.Button(root, text="",
font=("normal", 25), width=6, height=2,
command=lambda i=i: button_click(i)) for i in
range(9)]

for i, button in enumerate(buttons):
    button.grid(row=i//3, column=i%3)

current_player = "X"
label = tk.Label(root, text=f"Player:
{current_player}'s turn", font=("normal",
16))
label.grid(row=3, column=0, columnspan=3)

root.mainloop()
```

DESCRIPTION OR THESIS:

I am making a GUI based tic tac toe game in python.

What is tic tac toe game?

Tic tac toe is very simple and popular game in which two player uses "X" and "O". In this game the first player to get **three of your marks in a row**—either horizontally, vertically, or diagonally—on a 3x3 grid will win the game.

EXPLANATION OF GAME:

1. In first step, we will use tkinter module as an input which is a standard library in Python used for making GUI.
2. Now we will use messagebox using tkinter which is used to show any massage in case to show match draw for a draw game or win for a win.
3. Now we define a function "check_winner" which will check after every move if there is a winner or not. For this purpose we will make a list in which there will be 8 combinations to win a match.

"Combinations will be 3 horizontal, 3 vertical and 2 diagonal".

4. We will pass arguments when 3 boxes match with each other then that particular line will have a green background.

5. Here we will define a function 'button_click'. The purpose of this function is that when a user clicks on a button, the users symbol is displayed on that button. To achieve this, we will take an index parameter in the function, which will be the index of the button. Then we will check if the button is empty and if winner variable is false or not. The winner variable whether the game is still going on or not.

6. Here, we will define a function 'toggle_player' that will change the current player. For this we will use a global variable, 'current player', which will allow modification within the function. Then, we will alternate between 'X' and 'O' and this will continue.it will inform the user whose turn it is and which player is allowed to make a move.

7. Now we will create a root window using tkinter. We will name this window 'tic tac toe' and refer as "root". It will be used to create window and set title.

8. Here, we will create 9 buttons for the game whose font will be normal or a bit larger,

9. We will pass a lambda function in the command parameter, which will call the button click function and provide the index of that button.

A lambda function is an anonymous function that is written in a single line. These buttons will be stored in a list, which will be referred to as button.

10. To create these buttons, we will use a list comprehension, which is a python trick that allows us to create a list in a single list.

11. Now we will arrange these buttons in a single grid so that they appear in 3x3 layout. For this, we will use enumerate functions which gives us the index along with each button. Then we wil use the grid method to set the buttons in rows and columns. The rows and columns will be calculated based on the button's index.

12. Here, we will create a variable called current_player, which will held either 'X' or 'O'. we will start with 'X' and refer to it as current_player.

13. Now, we will create a variable called winner, which will start with false, and refer to it as winner. This variable will tell us whether the game is still ongoing or not.

14. Here, we will create a label digit which will inform the user about whose turn is it.

15. Now, we will arrange this label in the grid so that it appears below the buttons. Foe this, we will use the grid method and set the rows and columns to 3 , which will make the label stretch across the full width.

16. In the last, we will run the root window using the main loop method, which will display the window on the screen and handle user input.