

**A c# Project for Academic Year 2025 - 2026**

**Tic Tac Toe**

**Project**

Submitted in partial fulfillment of the requirement for the award of the degree

Bachelor of Application (BCA)

SubmittedBy : Vishal Parekh (92300527233)

: Neel Gardhariya (92300527211)

SubmittedTo:

Dr. Jignesh Hirapara

**DECLARATION**

I/Wehereby declare that this project work entitled **Tic Tac Toe Project** is a record done by me.

I also declare that the matter embodied in this project is genuine work done by me and has not been submitted whether to this University or to any other University / Institute for the fulfillment of the requirement of any course of study.

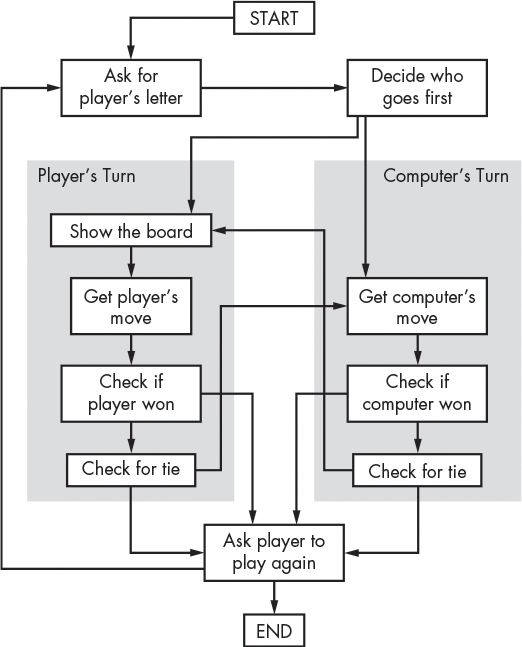
Place :University

Date : 15-04-2025

**Student Name-1:** (Enrollment No) **Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_**

**Student Name-2:** (Enrollment no) **Signature**: **\_\_\_\_\_\_\_\_\_\_\_\_\_**

Flow Chart Of Code:



**Table Of content:**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

| **Chapter** | **Particulars** |  |
| --- | --- | --- |
| **1** | Introduction |  |
| **2** | Objective of the Project |  |
| **3** | Existing System Overview |  |
| **4** | Hardware and Software Requirements |  |
| **5** | **System Design** |  |
|  | • Flowchart |  |
|  | • Use Case Diagram |  |
|  | • Class Diagram |  |
| **6** | Implementation (Code Overview) |  |
| **7** | Output Screens |  |
| **8** | Conclusion |  |
| **9** | Future Scope |  |
| **10** | Bibliography |  |
|  | **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** |  |
|  |  |  |

**Introduction to Project Defination :**

The Tic Tac Toe Game is a simple console based application designed to allow two players to play the classic game of Tic Tac Toe. The program is developed in C# using a modular structure and stores minimal data during execution. The system provides an interactive text-based interface for users to play the game, manage turns, and determine the winner.

**PREAMBLE**

**Module Description :**

The system is divided into the following modules:

* Game Board Module: Manages the 3x3 grid.
* Player Module: Handles player inputs and symbols (X or O).
* Game Logic Module: Checks win conditions and draw scenarios.
* Display Module: Renders the board and messages in the console.
* Replay Module: Allows players to restart the game after a match.

***Study Of Existing System :***

Manual versions of the Tic Tac Toe game

(on paper) are easy but lack features such as error checks and replay options. Existing digital versions may include ads or be too complex for beginners trying to learn programming.

***Inefficiency:***

* Manual play does not prevent illegal moves or check for win conditions automatically.
* Human errors can affect the fairness or flow of the game.

**Higher Probability of Human Error:**

* Mistakes in tracking turns or marking the board can lead to confusion.

**No Replay or Undo Options:**

* Manual gameplay lacks features to easily

**Restart or undo moves :**

* Limited Engagement:

Without a visual or interactive experience, manual play may feel static.

***TECHNICAL DESCRIPTION***

**Hardware Requirements:**

* Processor: Intel Core i3 or higher
* RAM: 2 GB or more
* Storage: Minimal (for code files)
* Input: Keyboard
* Display: Console output

***Software Requirements:***

* Windows OS
* .NET Framework
* Visual Studio / Visual Studio Code

***System Design And Development :***

**Diagrams as Applicable**

**Algorithm:**

1. Initialize game board
2. Display empty board
3. Repeat until game ends:
   * Accept player input
   * Update board
   * Check for win/draw
   * Switch player
4. Display result and ask for replay

***Database Design / File Structure:***

*No external files or database required for this project.*  
All data is managed in runtime memory using arrays and variables.

*Menu Design*

***A basic numbered menu allows the user to:***

1. Start Game
2. View Instructions
3. Exit

***Screen Design :***

***Console output showing board and options:***

**=== TIC TAC TOE GAME ===**

**1 | 2 | 3**

**-----------**

**4 | 5 | 6**

**-----------**

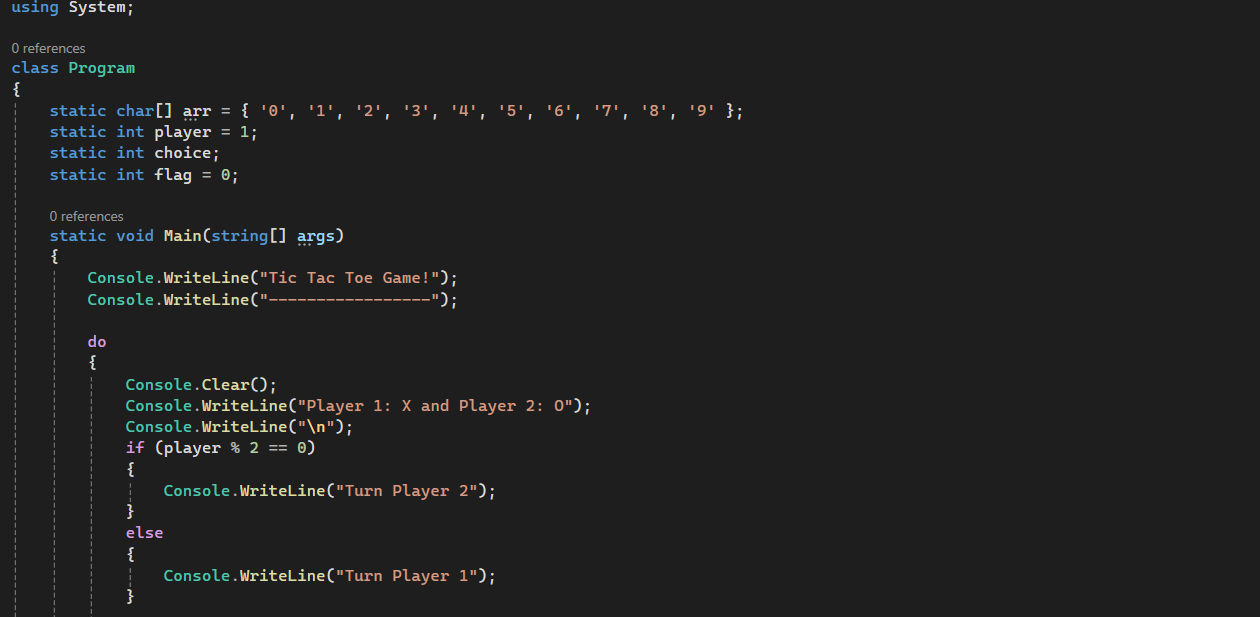
**7 | 8 | 9**

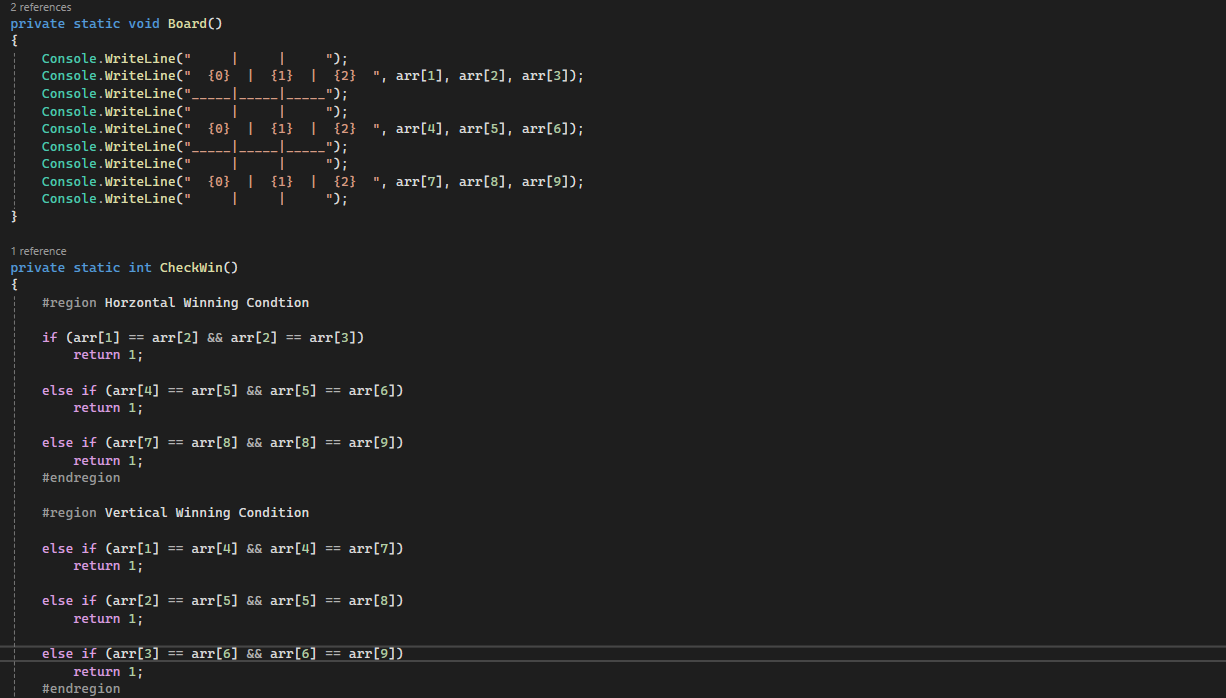
**Player X, choose your move:**

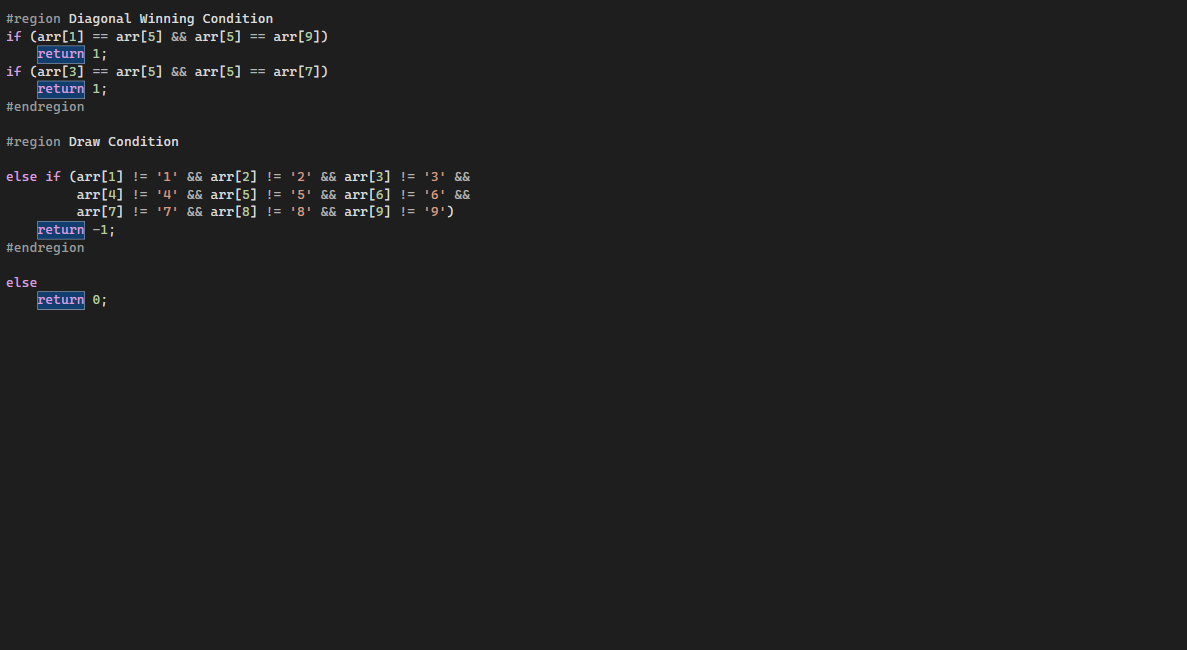
***Code of the Module***

**Main class: Program**

* Sub-methods: StartGame(), DisplayBoard(), PlayerMove(), CheckWinner(), ResetGame()
* Class: TicTacToeGame (contains boardand logic)

******

******

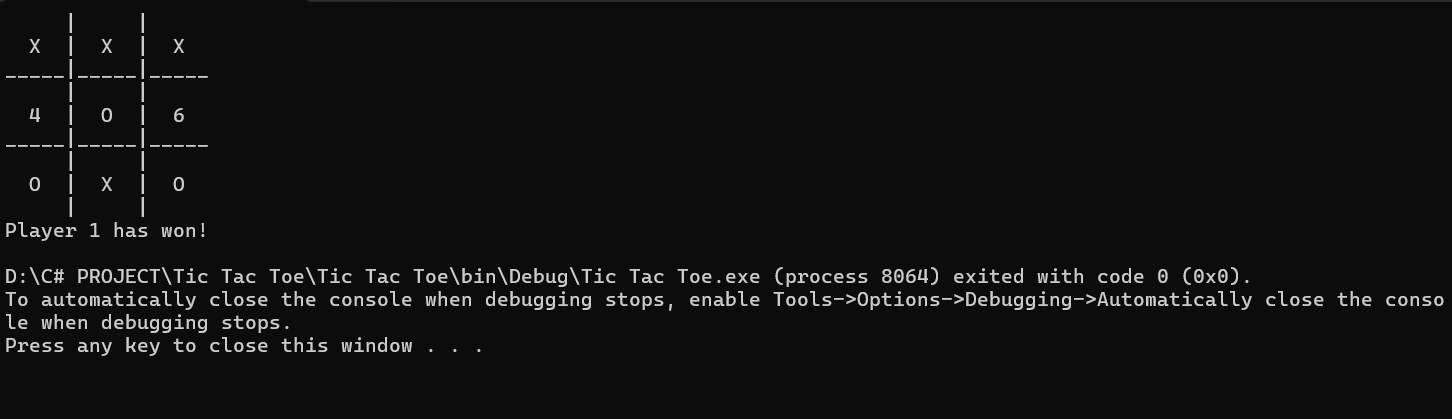
******

******

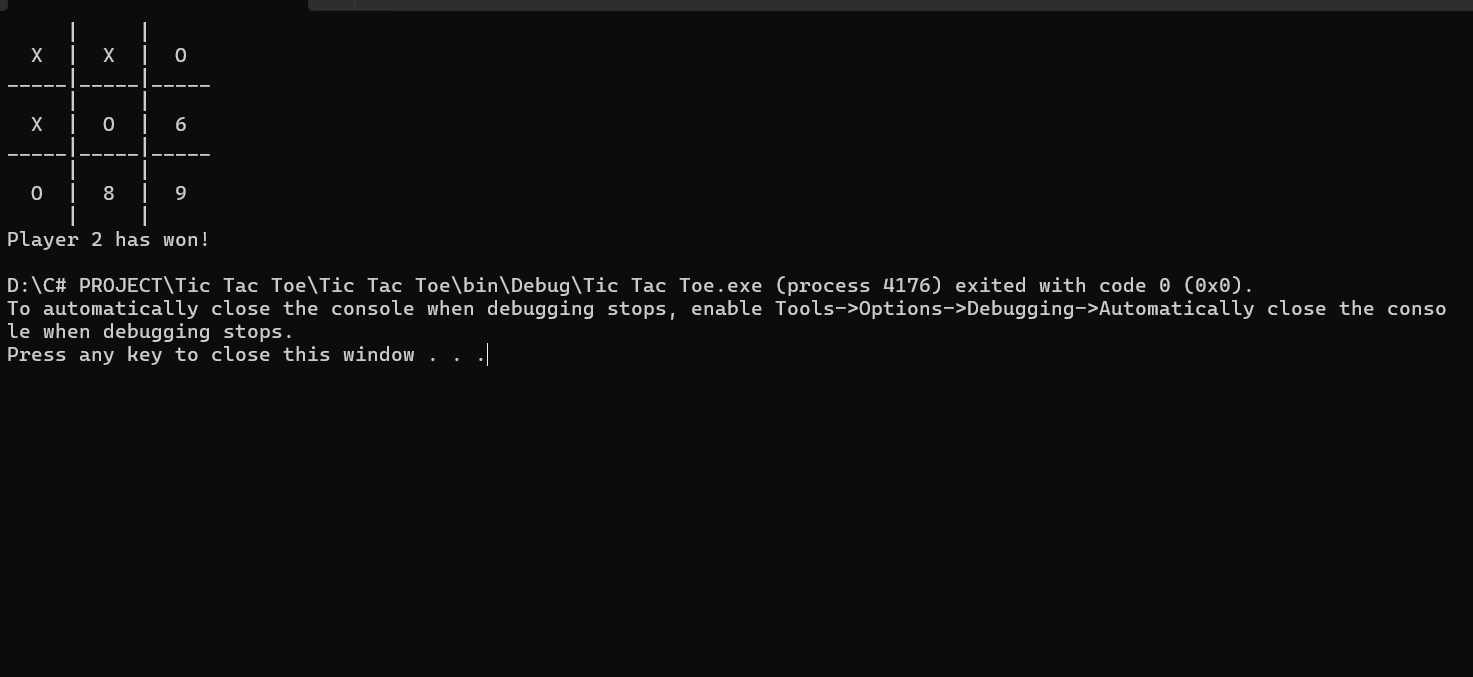
*“Format of the output”*

******

*“Player Can not choose repeated shell”*

******

*“Player 1 Has Won“*

******

*“Player 2 Has Won”*

***Conclusion :***

The Tic Tac Toe Game project provides a simple yet effective platform to learn programming concepts like arrays, loops, conditions, and modular structure. It enhances understanding of game logic implementation using C#.

***Learning During Project Work :***

* Fundamentals of game logic in C#
* Working with arrays and conditions
* User input validation
* Creating a menu-based console app
* Developing replay and reset functionalities

