*A*

*Laboratory Project Report*

*On a*

# TIC TAK TOE

*submitted in partial fulfillment of the requirements for the laboratory project in a*

### BACHELOR OF TECHNOLOGY

*in*

### ELECTRONICS AND COMMUNICATION ENGINEERING

*by*

A.SRINIVAS (22211A0415)

B.SIRI(22211A0440)

CH.KIRAN RAJ(22211A0447)

D.SRI VARDHAN (22211A0463)

*Under the guidance of*

**MS.M SHEREESHA**, Mtech,(Ph.D),

Assistant Professor



# B.V.RAJU INSTITUTE OF TECHNOLOGY

(UGC Autonomous, Accredited by NBA & NAAC) Vishnupur, Narspur, Medak(Dist.), Telangana State, India - 502313

2022 - 2023

## B. V. Raju Institute of Technology

(UGC Autonomous, Accredited By NBA & NAAC) Vishnupur, Narspur, Medak (Dist.), Telangana State, India – 502313

**CERTIFICATE**

This is to certify that the laboratory Project entitled **“STUDENT MANAGEMENT SYSTEM”**, being submitted by

#### SRINIVAS (22211A0415)

#### SIRI (22211A0440)

**KIRAN (22211A0447)**

**SRI VARDHAN (22211A0463)**

In partial fulfillment of the requirements for the laboratory project of BACHELOR OF TECHNOLOGY in **ELECTRONICS AND COMMUNICATION ENGINEERING** to B V RAJU

INSTITUTE OF TECHNOLOGY is a record of bonafide work carried out during a period from **November 2022 to March 2023** by them under the guidance of **Ms.SHEREESHA**, Assistant Professor, CSE Department.

This is to certify that the above statement made by the students is/are correct to the best of my knowledge.

#### Ms.M.SHEREESHA

Assistant Professor

## B. V. Raju Institute of Technology

(UGC Autonomous, Accredited By NBA & NAAC) Vishnupur, Narspur, Medak (Dist.), Telangana State, India – 502313

### CANDIDATE’S DECLARATION

I/We hereby certify that the work which is being presented in the project entitled **“TIC TAC TOE”** in partial fulfillment of the requirements for the project laboratory of Computational thinking and programming, work carried out during a period from **November 2022 to March 2023** under the guidance of **Ms.SHEREESHA**, Assistant Professor.

SRINIVAS(22211A0415) SIRI(22211A0440)

KIRAN(22211A0447)

SRI VARDHAN(22211A0463)

### ACKNOWLEDGEMENT

"I would like to express my sincere gratitude to the following individuals and institutions for their valuable contributions to the development of this **tic tac toe** using C language.

Firstly, I would like to thank my project supervisor Sai charan (btech) for his/her continuous support, guidance, and encouragement throughout the project. I am also grateful to Arun(btech) for their technical expertise and valuable insights that helped me overcome the challenges encountered during the development process.

I would like to acknowledge the support of  **BVRIT** for providing me with the necessary resources and facilities required for the successful development of this system.

Finally, I would like to thank all the members who participated in this project and provided valuable feedback to improve the system. Their feedback and suggestions were instrumental in the development of a user-friendly and efficient system.

Without the contributions of the above-mentioned individuals and institutions, this project would not have been possible."

SRINIVAS(22211A0415) SIRI(22211A0440)

KIRAN(22211A0447)

SRI VARDHAN(22211A0463)

### ABSTRACT

The Tic Tac Toe game is a classic two-player game played on a 3x3 grid. In this project, we have implemented Tic Tac Toe game using C programming language.

The game is played between two players, X and O, who take turns marking the spaces in a 3x3 grid. The player who succeeds in placing three of their marks in a horizontal, vertical, or diagonal row wins the game.

The program first displays the empty Tic Tac Toe board. Then, it asks the first player to choose either X or O. The program then accepts input from the players in the form of row and column numbers, indicating where they want to place their mark on the board. The program checks if the move is valid and updates the board accordingly.

The program continues to accept input from the players until one of the players wins the game or the game ends in a tie. After the game is over, the program displays the winner or the tie message and prompts the players if they want to play again.

The program uses functions to perform various tasks, such as checking for a win, checking if the board is full, and displaying the board.

Overall, this Tic Tac Toe game project implemented in C language demonstrates the basic concepts of game programming and logical problem-solving skills.

### CONTENTS:

**a)INTRODUCTION** **b)IMPLEMENTATION CODE** **c)RESULT**

**d)CONCLUSION** **e)REFERENCE**

#### INTRODUCTION:

Tic Tac Toe is a simple two-player game played on a 3x3 grid. Each player takes turns placing their symbol (typically "X" or "O") on the grid until one player gets three symbols in a row (horizontally, vertically, or diagonally), or the grid is completely filled without either player getting three in a row, resulting in a tie.

To implement Tic Tac Toe in C, you'll need to create a program that can display the game board, allow players to make their moves, check for a winner after each move, and handle ties.

Here are some of the main steps you could take to create a Tic Tac Toe program in C:

Create a 3x3 game board using a two-dimensional array.

Display the game board to the players using printf statements.

Prompt the first player to choose a cell on the board to place their symbol.

Update the game board with the first player's move.

Check if the first player has won the game or if the game is tied.

If the game is still ongoing, prompt the second player to choose a cell on the board to place their symbol.

Repeat steps 4-6 until a winner is declared or the game is tied.

Display the final game board and the outcome of the game.

You can use various C language constructs like loops, conditionals, and functions to make the implementation more effects and modular.

#### IMPLEMENTATION CODE

#include <stdio.h>

#include <conio.h>

char square[10] = { 'o', '1', '2', '3', '4', '5', '6', '7', '8', '9' };

int checkwin();

void board();

int main()

{

int player = 1, i, choice;

char mark;

do

{

board();

player = (player % 2) ? 1 : 2;

printf("Player %d, enter a number: ", player);

scanf("%d", &choice);

mark = (player == 1) ? 'X' : 'O';

if (choice == 1 && square[1] == '1')

square[1] = mark;

else if (choice == 2 && square[2] == '2')

square[2] = mark;

else if (choice == 3 && square[3] == '3')

square[3] = mark;

else if (choice == 4 && square[4] == '4')

square[4] = mark;

else if (choice == 5 && square[5] == '5')

square[5] = mark;

else if (choice == 6 && square[6] == '6')

square[6] = mark;

else if (choice == 7 && square[7] == '7')

square[7] = mark;

else if (choice == 8 && square[8] == '8')

square[8] = mark;

else if (choice == 9 && square[9] == '9')

square[9] = mark;

else

{

printf("Invalid move ");

player--;

getch();

}

i = checkwin();

player++;

}while (i == - 1);

board();

if (i == 1)

printf("==>\aPlayer %d win ", --player);

else

printf("==>\aGame draw");

getch();

return 0;

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

FUNCTION TO RETURN GAME STATUS

1 FOR GAME IS OVER WITH RESULT

-1 FOR GAME IS IN PROGRESS

O GAME IS OVER AND NO RESULT

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

int checkwin()

{

if (square[1] == square[2] && square[2] == square[3])

return 1;

else if (square[4] == square[5] && square[5] == square[6])

return 1;

else if (square[7] == square[8] && square[8] == square[9])

return 1;

else if (square[1] == square[4] && square[4] == square[7])

return 1;

else if (square[2] == square[5] && square[5] == square[8])

return 1;

else if (square[3] == square[6] && square[6] == square[9])

return 1;

else if (square[1] == square[5] && square[5] == square[9])

return 1;

else if (square[3] == square[5] && square[5] == square[7])

return 1;

else if (square[1] != '1' && square[2] != '2' && square[3] != '3' &&

square[4] != '4' && square[5] != '5' && square[6] != '6' && square[7]

!= '7' && square[8] != '8' && square[9] != '9')

return 0;

else

return - 1;

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

FUNCTION TO DRAW BOARD OF TIC TAC TOE WITH PLAYERS MARK

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

void board()

{

system("cls");

printf("\n\n\tTic Tac Toe\n\n");

printf("Player 1 (X) - Player 2 (O)\n\n\n");

printf(" | | \n");

printf(" %c | %c | %c \n", square[1], square[2], square[3]);

printf("\_\_\_\_\_|\_\_\_\_\_|\_\_\_\_\_\n");

printf(" | | \n");

printf(" %c | %c | %c \n", square[4], square[5], square[6]);

printf("\_\_\_\_\_|\_\_\_\_\_|\_\_\_\_\_\n");

printf(" | | \n");

printf(" %c | %c | %c \n", square[7], square[8], square[9]);

printf(" | | \n\n");

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

END OF PROJECT

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

**C) RESULTS:**

Tic Tac Toe

Player 1 (X) - Player 2 (O)

| |

1 | 2 | 3

\_\_|\_|\_\_

| |

4 | 5 | 6

\_\_|\_|\_\_

| |

7 | 8 | 9

| |

Player 1, enter a number: 1

sh: 1: cls: not found

Tic Tac Toe

Player 1 (X) - Player 2 (O)

| |

X | 2 | 3

\_\_|\_|\_\_

| |

4 | 5 | 6

\_\_|\_|\_\_

| |

7 | 8 | 9

| |

Player 2, enter a number: 7

sh: 1: cls: not found

Tic Tac Toe

Player 1 (X) - Player 2 (O)

| |

X | 2 | 3

\_\_|\_|\_\_

| |

4 | 5 | 6

\_\_|\_|\_\_

| |

O | 8 | 9

| |

Player 1, enter a number: 2

sh: 1: cls: not found

Tic Tac Toe

Player 1 (X) - Player 2 (O)

| |

X | X | 3

\_\_|\_|\_\_

| |

4 | 5 | 6

\_\_|\_|\_\_

| |

O | 8 | 9

| |

Player 2, enter a number: 3

sh: 1: cls: not found

Tic Tac Toe

Player 1 (X) - Player 2 (O)

| |

X | X | O

\_\_|\_|\_\_

| |

4 | 5 | 6

\_\_|\_|\_\_

| |

O | 8 | 9

| |

Player 1, enter a number: 4

sh: 1: cls: not found

Tic Tac Toe

Player 1 (X) - Player 2 (O)

| |

X | X | O

\_\_|\_|\_\_

| |

X | 5 | 6

\_\_|\_|\_\_

| |

* | 8 | 9

| |

Player 2, enter a number: 5

sh: 1: cls: not found

Tic Tac Toe

Player 1 (X) - Player 2 (O)

| |

X | X | O

\_\_|\_|\_\_

| |

X | O | 6

\_\_|\_|\_\_

| |

O | 8 | 9

| |

==>Player 2 win

**D)CONCLUSION:**

**In conclusion**,

1.Tic Tac Toe is a classic game that is relatively simple to implement in C. It can be a great project for beginners who are looking to improve their programming skills.

2.The basic structure of a Tic Tac Toe program in C involves creating a 3x3 array to represent the game board, and then using loops and conditional statements to allow the players to make moves and check for a winner.

3.Depending on the complexity of the project, there are a number of different features that could be added to make the game more interesting or challenging. For example, you could create an AI opponent using techniques like minimax or alpha-beta pruning.

4.When writing a Tic Tac Toe program in C, it is important to pay close attention to details like error handling and input validation to ensure that the program is robust and user-friendly.

Overall, a Tic Tac Toe project in C can be a fun and rewarding way to practice programming skills and explore the basics of game development.

**E) References**

* 1. THE C PROGRAMMING LANGUAGE [Brian Kernighan](http://en.wikipedia.org/wiki/Brian_Kernighan); [Dennis Ritchie](http://en.wikipedia.org/wiki/Dennis_Ritchie)

sanfoundry.

blogspot