TIC-TAC-TOE GAME

Submitted by: -

Name of the Student: Souraj Chatterjee

Enrollment number:12022002004074

Registration number: 221040110821

Section: J

Class roll number:62

Stream: INFORMATION TECHNOLOGY (IT)

Subject: Programming for Problem Solving using C

Subject Code: ESC103 (Pr.)

Department: Basic Science and Humanities (BSH)

Under the supervision of: -

Dr. Swarnendu Ghosh

Academic Year: 2022-26

PROJECT REPORT SUBMITTED IN PARTIAL FULFILLMENT OF THE
REQUIREMENTS FOR THE FIRST SEMESTER

DEPARTMENT OF BASIC SCIENCE AND HUMANITIES

INSTITUTE OF ENGINEERING AND MANAGEMENT,

KOLKATA

CERTIFICATE OF RECOMMENDATION

We hereby recommend that the project prepared under our supervision by **Souraj Chatterjee**, entitled "**Tic-Tac-Toe game**" be accepted in improving the ability to concentrate and focus as well as developing problem-solving skils.

_____   _____

Head of the Department   Project Supervisor
Basic Science and Humanities
IEM, Kolkata

1. Introduction:

This project is assigned to me for developing a 'Tic-Tac-Toe Game' using C language. It is the game for two players. One player plays X and the other plays O. The players take turns placing their marks on a grid of three-by-three cells. If a given player gets three marks in a row horizontally, vertically, or diagonally, then that player wins the game.

2. Variable Description:

The different variables used in the project are listed under: -

1. int- To store integer datatypes.
2. char- To store character datatypes.

3. Function Description:

The different functions(structures) used in this project are listed under: -

1. Function to display the table board
2. Function to insert an 'X' or 'O' into the table
3. Function to check if anyone has won
4. Function to check for a draw
5. Function to play the match

4. Programs:

Tic-Tac-Toe Game.c

```c
#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;
```

```c
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;
```

```c
        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;
}


int checkwin()
```

```c
{
    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
```

```c
        return  - 1;

}


void board()
{

    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");
}
```
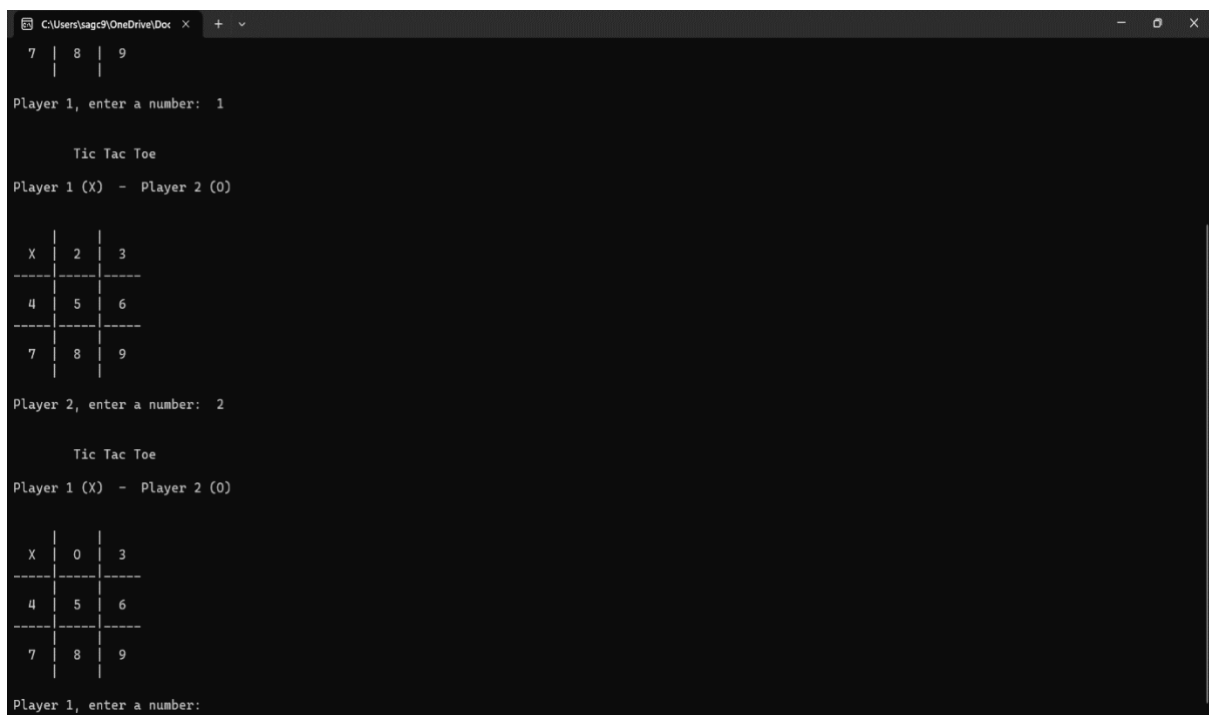
5. Outputs:

Sample outputs(screenshots) to demonstrate the functionalities in programs   are
listed below

```
7 | 8 | 9
  |   |

Player 1, enter a number:  5

        Tic Tac Toe

Player 1 (X)  -  Player 2 (O)


  |   |
X | O | 3
-----|-----|-----
  |   |
4 | X | 6
-----|-----|-----
  |   |
7 | 8 | 9
  |   |

Player 2, enter a number:  3

        Tic Tac Toe

Player 1 (X)  -  Player 2 (O)


  |   |
X | O | O
-----|-----|-----
  |   |
4 | X | 6
-----|-----|-----
  |   |
7 | 8 | 9
  |   |

Player 1, enter a number:  |
```

```
 7 | 8 | 9
   |   |

Player 1, enter a number:  1


        Tic Tac Toe

Player 1 (X)  -  Player 2 (0)


   |   |
 X | 2 | 3
___|___|___
   |   |
 4 | 5 | 6
___|___|___
   |   |
 7 | 8 | 9
   |   |

Player 2, enter a number:  2


        Tic Tac Toe

Player 1 (X)  -  Player 2 (0)


   |   |
 X | 0 | 3
___|___|___
   |   |
 4 | 5 | 6
___|___|___
   |   |
 7 | 8 | 9
   |   |

Player 1, enter a number:
```

```
 7 | 8 | 9
   |   |

Player 2, enter a number:  3

        Tic Tac Toe

Player 1 (X)  -  Player 2 (0)


   |   |
 X | 0 | 0
___|___|___
   |   |
 4 | X | 6
___|___|___
   |   |
 7 | 8 | 9
   |   |

Player 1, enter a number:  9

        Tic Tac Toe

Player 1 (X)  -  Player 2 (0)


   |   |
 X | 0 | 0
___|___|___
   |   |
 4 | X | 6
___|___|___
   |   |
 7 | 8 | X
   |   |

==>Player 1 win
```

```
    7 | 8 | 9
      |   |

Player 1, enter a number:  4

         Tic Tac Toe

Player 1 (X)  -  Player 2 (0)


      |   |
   X  | 0 | 3
 _____|_____|_____
      |   |
   X  | 5 | 6
 _____|_____|_____
      |   |
   7  | 8 | 9
      |   |

Player 2, enter a number:  7

         Tic Tac Toe

Player 1 (X)  -  Player 2 (0)


      |   |
   X  | 0 | 3
 _____|_____|_____
      |   |
   X  | 5 | 6
 _____|_____|_____
      |   |
   0  | 8 | 9
      |   |

Player 1, enter a number:  |
```

```
O | 8 | X
  |   |

Player 1, enter a number:  8

        Tic Tac Toe

Player 1 (X)  -  Player 2 (O)


    |   |
  X | O | 3
____|____|____
    |   |
  X | O | 6
____|____|____
    |   |
  O | X | X
    |   |

Player 2, enter a number:  3

        Tic Tac Toe

Player 1 (X)  -  Player 2 (O)


    |   |
  X | O | O
____|____|____
    |   |
  X | O | 6
____|____|____
    |   |
  O | X | X
    |   |

==>Player 2 win
```

```
7 | 0 | X
  |   |

Player 2, enter a number:  4


        Tic Tac Toe

Player 1 (X)  -  Player 2 (0)


    |   |
 X  | X | 0
____|____|____
    |   |
 0  | 0 | X
____|____|____
    |   |
 7  | 0 | X
    |   |

Player 1, enter a number:  7


        Tic Tac Toe

Player 1 (X)  -  Player 2 (0)


    |   |
 X  | X | 0
____|____|____
    |   |
 0  | 0 | X
____|____|____
    |   |
 X  | 0 | X
    |   |

==>Game draw
```