MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE, GWALIOR

(A Govt. Aided UGC Autonomous & NAAC Accredited Institute Affiliated to RGPV, Bhopal)

Skills Based Mini Project Report

On

Tic Tac Toe Game

Submitted by:

PANKAJ MAHOR

(0901CA211035)

Mentor:

Dr. Anshu Chaturvedi

(Professor)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE
GWALIOR - 474005 (MP) est. 1957

(July-Dec 2021)

MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE, GWALIOR

(A Govt. Aided UGC Autonomous & NAAC Accredited Institute Affiliated to RGPV, Bhopal)

CERTIFICATE

This is certified that **Pankaj Mahor (0901CA211035)** has submitted the project report titled **Tic Tac Toe Game** under the mentorship of **Dr. Anshu Chaturvedi** (Professor & Faculty coordinator),as the skills based mini project in 1st year of Master of Computer Application in Computer Science and Engineering from Madhav Institute of Technology and Science, Gwalior. M.P.

Dr. Anshu Chaturvedi
Faculty coordinator & professor
Computer sicence and engineering

MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE, GWALIOR

(A Govt. Aided UGC Autonomous & NAAC Accredited Institute Affiliated to RGPV, Bhopal)

DECLARATION

I hereby declare that the work being presented in this project report, for the fulfilment of partial requirement of the skills based mini project in 1st year of Master of Computer Application in Computer Science and Engineering at **Madhav Institute of Technology & Science,Gwalior** is an authenticated and original record of my work under the mentorship of **Dr. Anshu chaturvedi** (Professor), MITS Gwalior.

I declare that I have not submitted the matter embodied in this report any where else.

Pankaj Mahor

PANKAJ MAHOR
(0901CA211035)
1st Year,
Master of Computer Application,
Computer Science and Engineering

ACKNOWLEDGEMENT

The full semester project has proved to be pivotal to my career. I am thankful to my institute, **Madhav Institute of Technology and Science** to allow me to continue my disciplinary project. I extend my gratitude to the Director of the institute, **Dr. R. K. Pandit** and Dean Academics, **Dr. Manjaree Pandit** for this.

I would sincerely like to thank my department, **Department of Computer Science and Engineering**, for allowing me to explore this project. I humbly thank **Dr. Manish Dixit**, Professor and Head, Department of Computer Science and Engineering, for his continued support during the course of this engagement, which eased the process and formalities involved.

I am sincerely thankful to my faculty coordinator. I am grateful to the guidance of **Dr. Anshu chaturvedi**, (Professor), Computer Science and Engineering, for her continued support and guidance throughout the project. I am also very thankful to the faculty and staff of the department.

Pankaj Mahor

PANKAJ MAHOR
(0901CA211035)
1st Year,
Master of Computer Application,
Computer Science and Engineering

ABSTRACT

The game of Tic-tac-toe is one of the most commonly known games. This game does not allow one to win all the time and a significant proportion of games played results in a draw. Thus, the The best a player can hope for is to not lose the game. This study is aimed at evolving a number of no- loss strategies using genetic algorithms and comparing them with existing methodologies.To efficiently evolve no-loss strategies,

we have developed innovative ways of representing and evaluating a solution, initializing the GA population, developing GA operators including an elite preserving scheme. Interestingly, our GA implementation is able to find more than 72 thousand no-loss strategies for playing the game.

Moreover, an analysis of these solutions has given us insights about how toplay the game to not lose it. Based on this experience, we have developed specialized efficient strategies having a high win-to-draw ratio. The study and its results are interesting and can be encouraging for the techniques to be applied to other board games for finding efficient strategies.

CONTENTS

1. INTRODUCTION

Tic-tac-toe is a game in which two player pick X's or O' s alternately in a matrix cell formed by two vertical and horizontal lines crossing each other each player triesn to place Three of their marks in a horizontal,vertical, or diagnonal row. The player who put in this from will succeed in the game. In this article, we'll write a program to player **Tic-tac-toe** game in C language.

The tic-tac-toe game is played on a 3 x 3 grid the game is player by two player, who take turns, the first player marks moves with a circle, the second with a cross. The player who Has formed a hrizontal,vertical,or sequence of three marks wins. Your program should draw the game board,as the use for the coordinates of the next mark, change the players after every successful move, and pronounce the winner. Our project name is Tic-tac-toe game. This game is very popular and is fairly simple by it self. It is actually a two player game .In this game , there is a boardwith n x n squares. In our game, it is 3 x 3 squares. Th e goal of Tic-Tac-Toe is to be one of the player s to get thre e same symbols in arrow - horizontally,vertically or diagonally - on a 3 x 3 grid.. Tic-tac-toe is played on a three-by-three grid by two players, who alternately place the marks X and O in one of the nine spaces in the grid.

Tic-tac-toe , also called **noughts and crosses** and many other names, is a **paper and pencil game** between two players, **O** and **X**, who alternate in marking the spaces in a 3×3 board. A player wins by getting three of their own marks in a horizontal, vertical or diagonal row .This game are 3x3 grid In digital format in playing.

2.Objectives of the Gaming System

The game is developed for full-time entertainment and enthusiasms. It teaches the Gamer to be alert at every situation he /she faces, because if the Gamer is not fully alert and notice the saucer fire he / she must be hit by the Saucer bombs. Though the proposed game is an Action game, it doesn’t involve direct violence. No zombie killing, animal kills or human killing is performed in the game. So it can also be viewed as a non-violence game. Kids can also play this game, because the design of the game is very simple, controlling the game is very easy. The game are easy toplayed by to people , two small child.

3.Program Explanation

1.Consider a board with the nine positions numbered as follows................>

1	2	3
4	5	6
7	8	9

2.The program has the following functions which are performing small tasks.

3.1 NitializeBoard() Function

This method is used to initialize the tic tac toe game board. The initialized board is given below.

```
   |   |

   |   |

   |   |
```

3.2 ShowBoard() Function

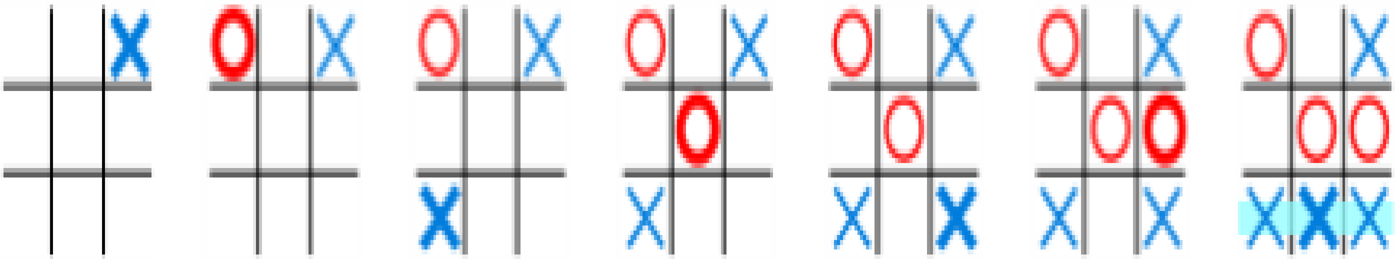
showBoard() method is used to print the game board after every play of each player.

```
  O  |     |                 O  |     |
________________            ________________
     |     |                    |  X  |
________________            ________________
     |     |                    |     |

  Player 1 Move               Player 2 Moves
```

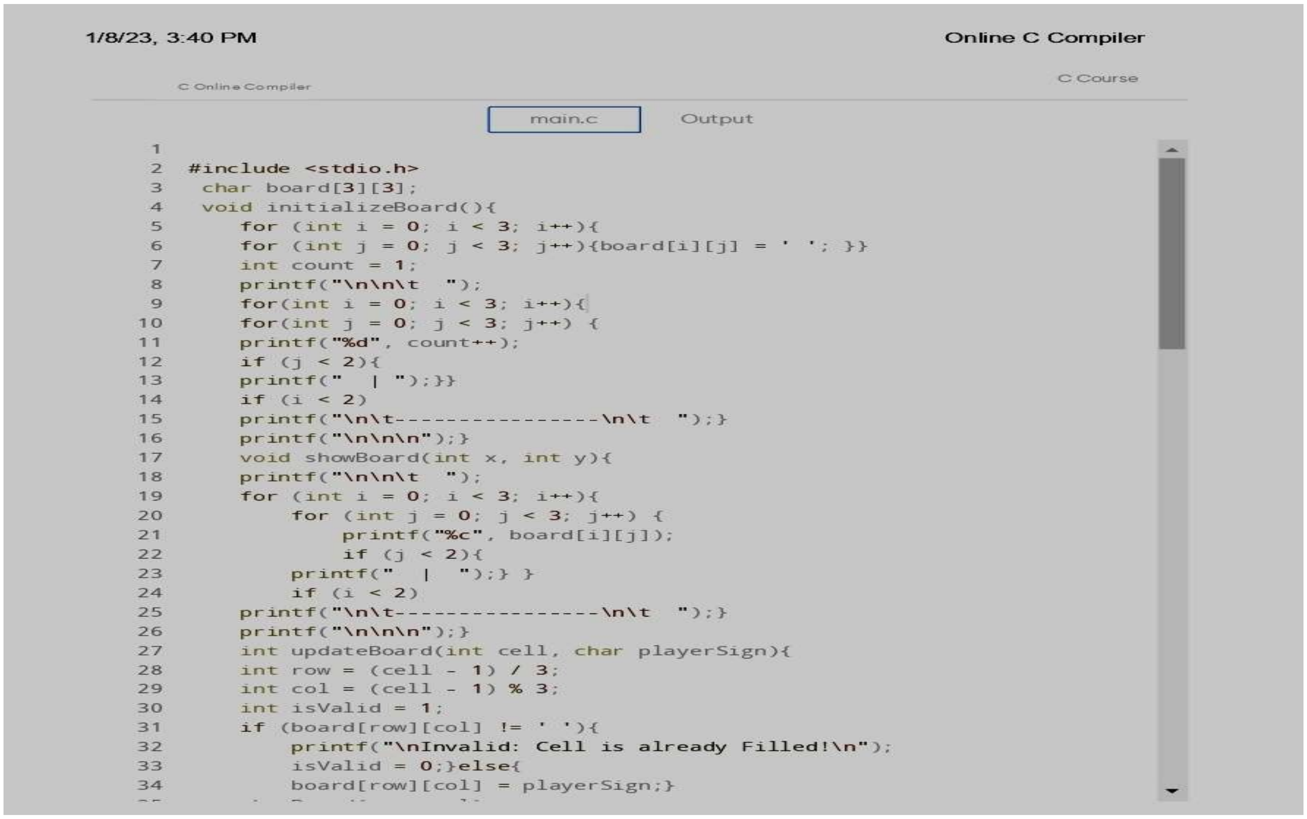
3.3 CheckWinner()

checkWinner() method used to find the winner of the game. It checks on each row, column, and diagonal. If the same sign is present in any one of these places then print the winner

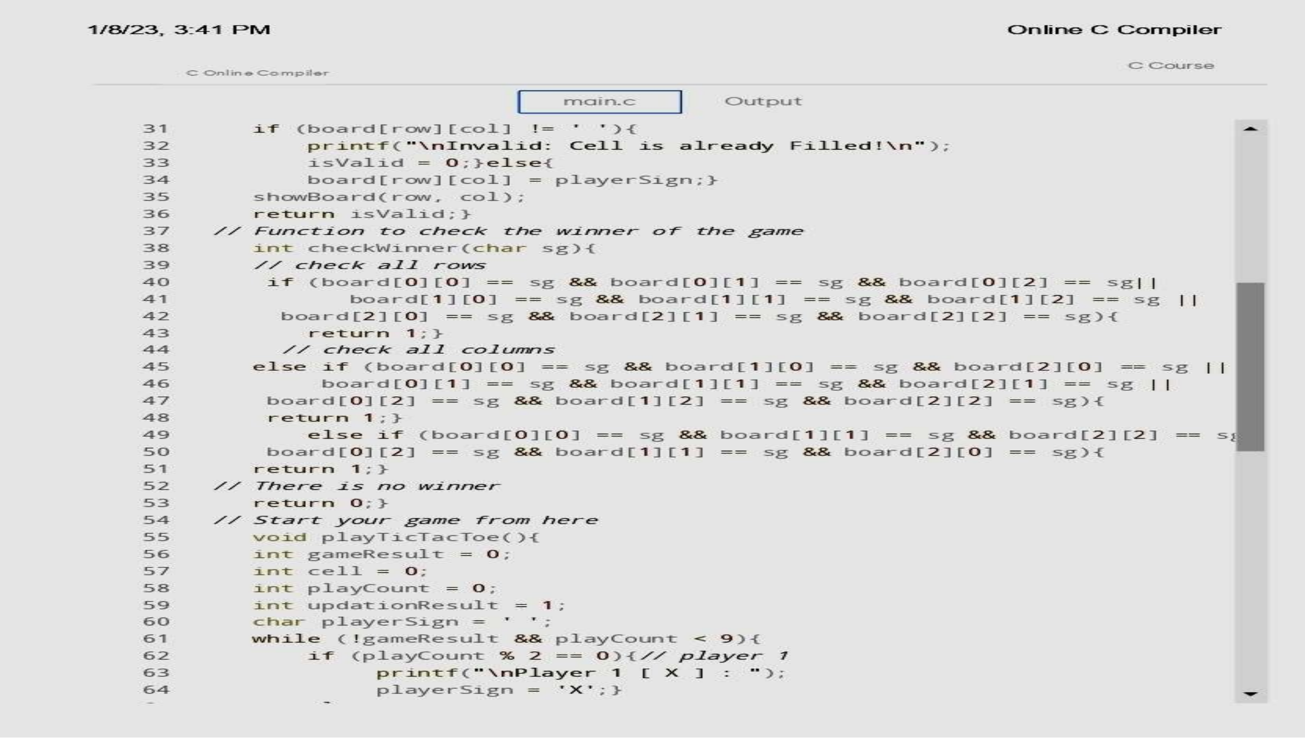
Games are playing, In the following example, the first player (*X*) wins the game in seven steps:



4. CODING SCREENSHOTS

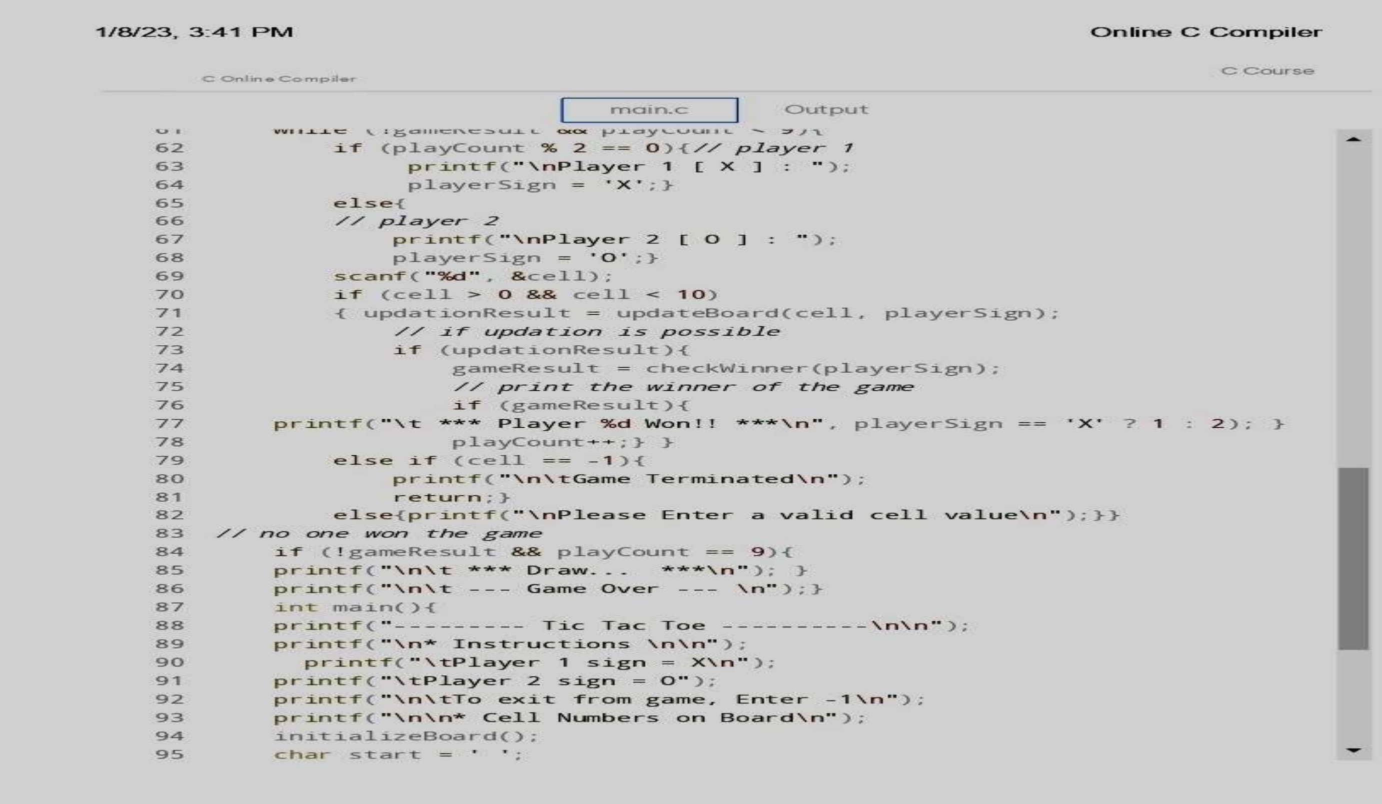


```c

#include <stdio.h>
 char board[3][3];
 void initializeBoard(){
    for (int i = 0; i < 3; i++){
    for (int j = 0; j < 3; j++){board[i][j] = ' '; }}
    int count = 1;
    printf("\n\n\t  ");
    for(int i = 0; i < 3; i++){
    for(int j = 0; j < 3; j++) {
    printf("%d", count++);
    if (j < 2){
    printf("  | ");}}
    if (i < 2)
    printf("\n\t-----------------\n\t  ");}
    printf("\n\n\n");}
    void showBoard(int x, int y){
    printf("\n\n\t  ");
    for (int i = 0; i < 3; i++){
        for (int j = 0; j < 3; j++) {
            printf("%c", board[i][j]);
            if (j < 2){
        printf("  |  ");} }
        if (i < 2)
    printf("\n\t-----------------\n\t  ");}
    printf("\n\n\n");}
    int updateBoard(int cell, char playerSign){
    int row = (cell - 1) / 3;
    int col = (cell - 1) % 3;
    int isValid = 1;
    if (board[row][col] != ' '){
        printf("\nInvalid: Cell is already Filled!\n");
        isValid = 0;}else{
        board[row][col] = playerSign;}
```

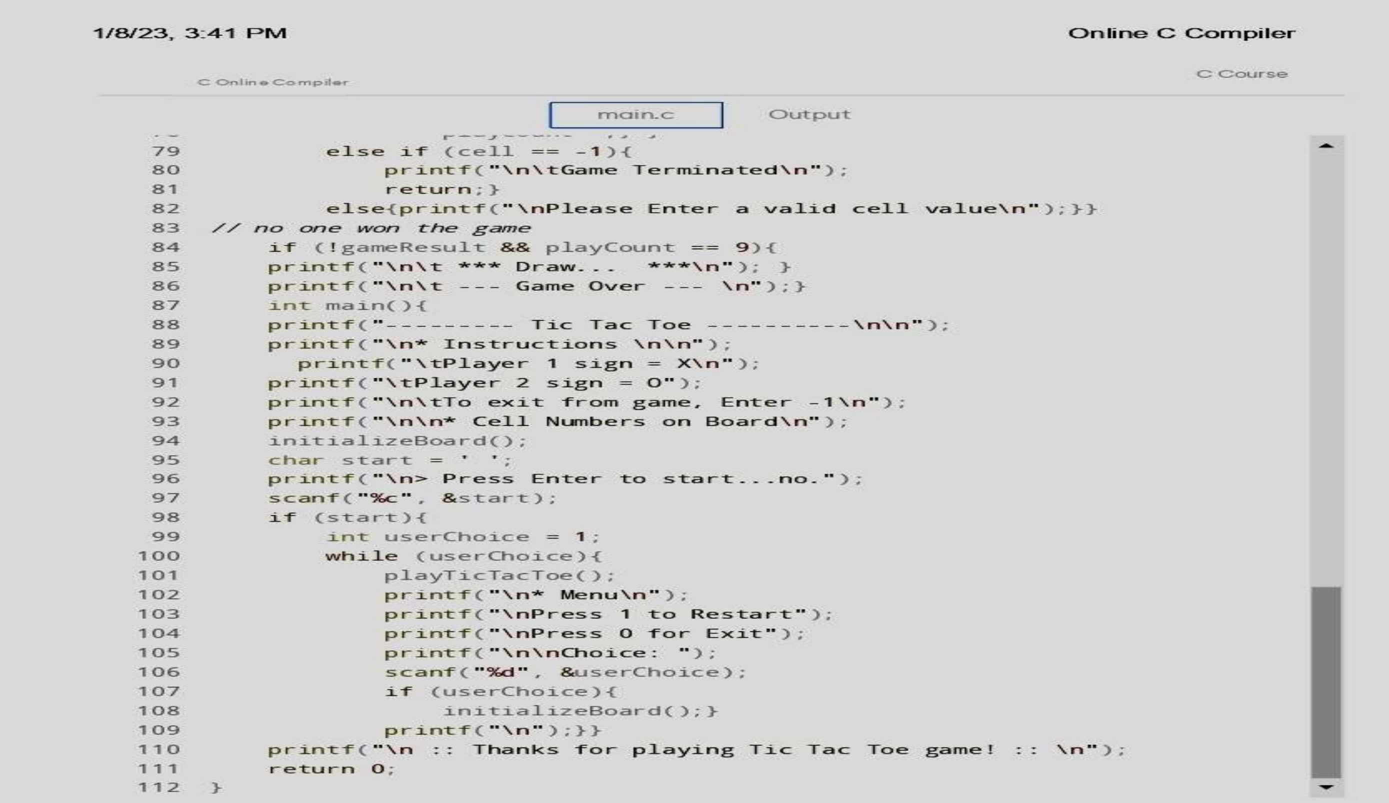


```c
    if (board[row][col] != ' '){
        printf("\nInvalid: Cell is already Filled!\n");
        isValid = 0;}else{
        board[row][col] = playerSign;}
    showBoard(row, col);
    return isValid;}
 // Function to check the winner of the game
    int checkWinner(char sg){
    // check all rows
     if (board[0][0] == sg && board[0][1] == sg && board[0][2] == sg||
          board[1][0] == sg && board[1][1] == sg && board[1][2] == sg ||
      board[2][0] == sg && board[2][1] == sg && board[2][2] == sg){
        return 1;}
      // check all columns
    else if (board[0][0] == sg && board[1][0] == sg && board[2][0] == sg ||
         board[0][1] == sg && board[1][1] == sg && board[2][1] == sg ||
     board[0][2] == sg && board[1][2] == sg && board[2][2] == sg){
     return 1;}
        else if (board[0][0] == sg && board[1][1] == sg && board[2][2] == sg
     board[0][2] == sg && board[1][1] == sg && board[2][0] == sg){
    return 1;}
 // There is no winner
    return 0;}
 // Start your game from here
    void playTicTacToe(){
    int gameResult = 0;
    int cell = 0;
    int playCount = 0;
    int updationResult = 1;
    char playerSign = ' ';
    while (!gameResult && playCount < 9){
        if (playCount % 2 == 0){// player 1
            printf("\nPlayer 1 [ X ] : ");
            playerSign = 'X';}
```

4. CODING SCREENSHOTS



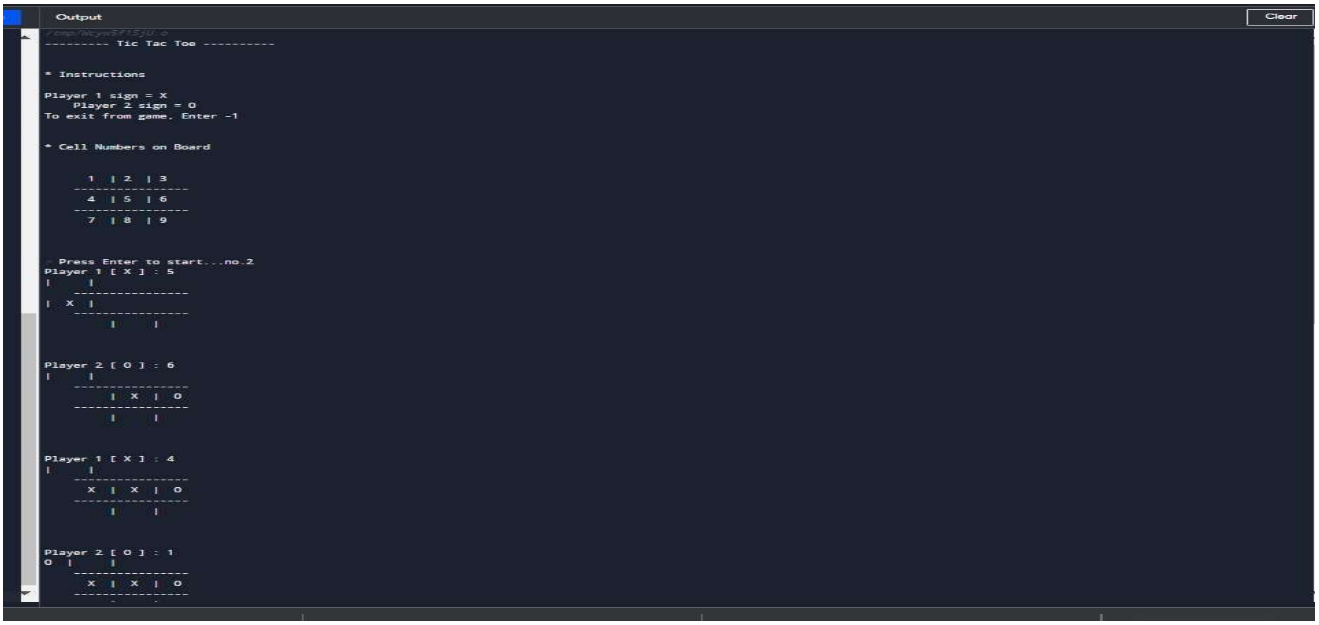
```
    while (!gameResult && playCount < 9){
        if (playCount % 2 == 0){// player 1
             printf("\nPlayer 1 [ X ] : ");
             playerSign = 'X';}
        else{
        // player 2
            printf("\nPlayer 2 [ O ] : ");
            playerSign = 'O';}
        scanf("%d", &cell);
        if (cell > 0 && cell < 10)
        { updationResult = updateBoard(cell, playerSign);
            // if updation is possible
            if (updationResult){
                gameResult = checkWinner(playerSign);
                // print the winner of the game
                if (gameResult){
    printf("\t *** Player %d Won!! ***\n", playerSign == 'X' ? 1 : 2); }
                playCount++;} }
        else if (cell == -1){
            printf("\n\tGame Terminated\n");
            return;}
        else{printf("\nPlease Enter a valid cell value\n");}}
// no one won the game
    if (!gameResult && playCount == 9){
    printf("\n\t *** Draw...  ***\n"); }
    printf("\n\t --- Game Over --- \n");}
    int main(){
    printf("--------- Tic Tac Toe ----------\n\n");
    printf("\n* Instructions \n\n");
      printf("\tPlayer 1 sign = X\n");
    printf("\tPlayer 2 sign = O");
    printf("\n\tTo exit from game, Enter -1\n");
    printf("\n\n* Cell Numbers on Board\n");
    initializeBoard();
    char start = ' ';
```



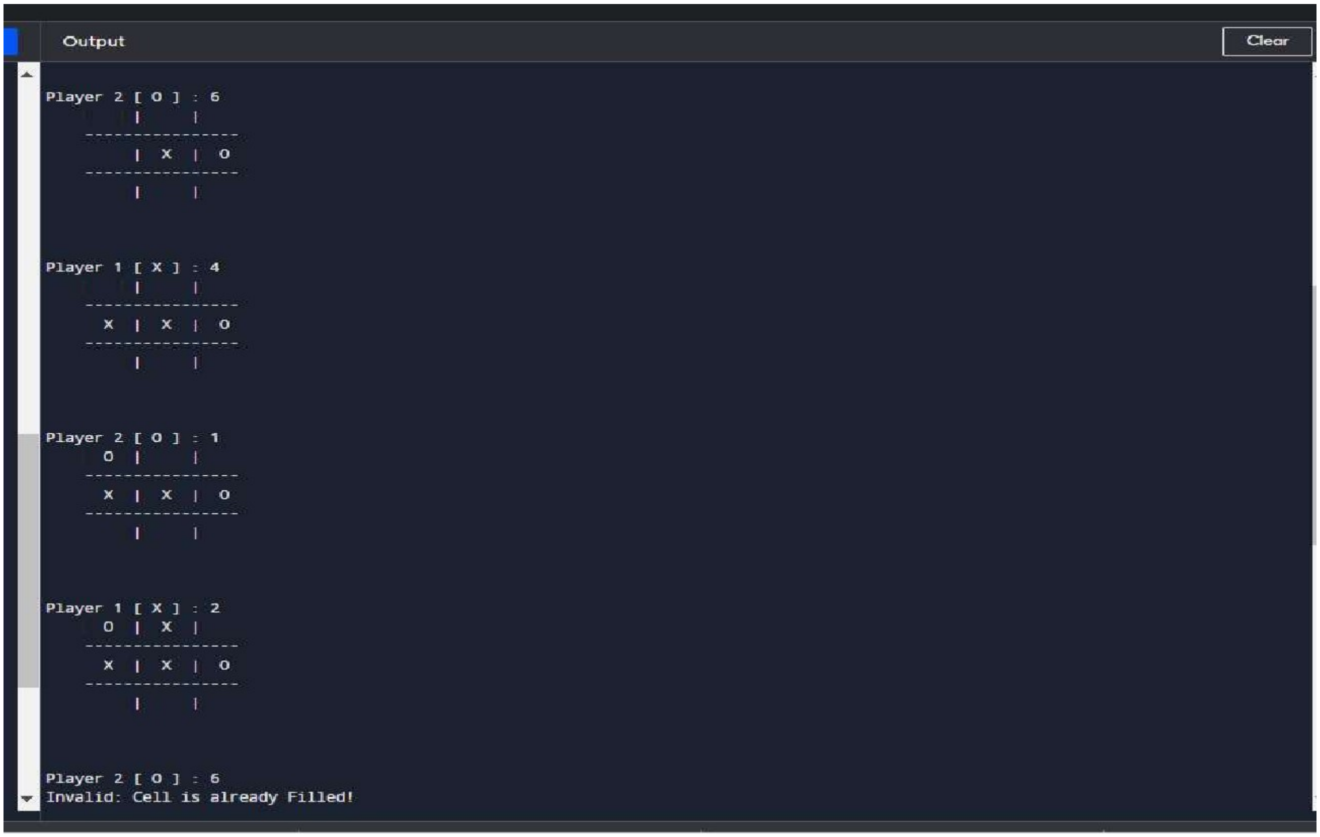
```
        else if (cell == -1){
            printf("\n\tGame Terminated\n");
            return;}
        else{printf("\nPlease Enter a valid cell value\n");}}
// no one won the game
    if (!gameResult && playCount == 9){
    printf("\n\t *** Draw...  ***\n"); }
    printf("\n\t --- Game Over --- \n");}
    int main(){
    printf("--------- Tic Tac Toe ----------\n\n");
    printf("\n* Instructions \n\n");
      printf("\tPlayer 1 sign = X\n");
    printf("\tPlayer 2 sign = O");
    printf("\n\tTo exit from game, Enter -1\n");
    printf("\n\n* Cell Numbers on Board\n");
    initializeBoard();
    char start = ' ';
    printf("\n> Press Enter to start...no.");
    scanf("%c", &start);
    if (start){
        int userChoice = 1;
        while (userChoice){
            playTicTacToe();
            printf("\n* Menu\n");
            printf("\nPress 1 to Restart");
            printf("\nPress 0 for Exit");
            printf("\n\nChoice: ");
            scanf("%d", &userChoice);
            if (userChoice){
                initializeBoard();}
            printf("\n");}}
    printf("\n :: Thanks for playing Tic Tac Toe game! :: \n");
    return 0;
}
```

5. OUTPUT SCREENSHOTS

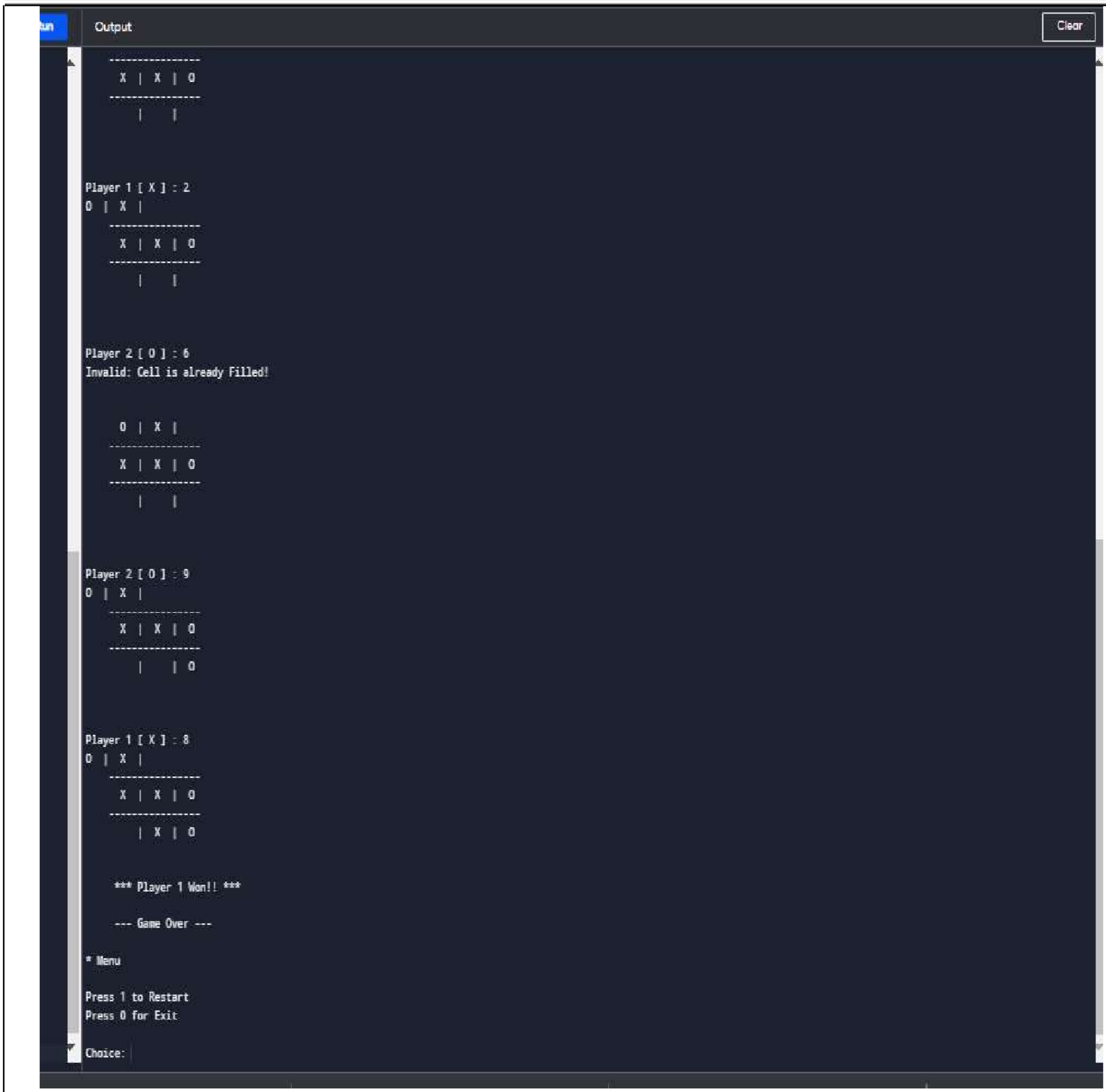
Tic-Tac-Toe game & Run



```
/tmp/Wcyw5f15jU.o
--------- Tic Tac Toe ----------


* Instructions

Player 1 sign = X
     Player 2 sign = O
To exit from game, Enter -1


* Cell Numbers on Board


     1  | 2  | 3
   ---------------
     4  | 5  | 6
   ---------------
     7  | 8  | 9



> Press Enter to start...no.2
Player 1 [ X ] : 5
|     |
   ---------------
|  X  |
   ---------------
        |     |



Player 2 [ O ] : 6
|     |
   ---------------
        |  X  |  O
   ---------------
        |     |



Player 1 [ X ] : 4
|     |
   ---------------
     X  |  X  |  O
   ---------------
        |     |



Player 2 [ O ] : 1
O  |     |
   ---------------
     X  |  X  |  O
   ---------------
```



```
Player 2 [ O ] : 6
        |     |
   ----------------
        |  X  |  O
   ----------------
        |     |



Player 1 [ X ] : 4
        |     |
   ----------------
     X  |  X  |  O
   ----------------
        |     |



Player 2 [ O ] : 1
     O  |     |
   ----------------
     X  |  X  |  O
   ----------------
        |     |



Player 1 [ X ] : 2
     O  |  X  |
   ----------------
     X  |  X  |  O
   ----------------
        |     |



Player 2 [ O ] : 6
Invalid: Cell is already Filled!
```



```
    ----------------
      X  |  X  |  0
    ----------------
         |     |



Player 1 [ X ] : 2
0  |  X  |
    ----------------
      X  |  X  |  0
    ----------------
         |     |



Player 2 [ 0 ] : 6
Invalid: Cell is already Filled!


      0  |  X  |
    ----------------
      X  |  X  |  0
    ----------------
         |     |



Player 2 [ 0 ] : 9
0  |  X  |
    ----------------
      X  |  X  |  0
    ----------------
         |     |  0



Player 1 [ X ] : 8
0  |  X  |
    ----------------
      X  |  X  |  0
    ----------------
         |  X  |  0


     *** Player 1 Won!! ***

     --- Game Over ---

* Menu

Press 1 to Restart
Press 0 for Exit

Choice:
```

6.CONCLUSION

The Tic Tac Toe game is most familiar among all the age groups. Intelligence can be property of any purpose-driven decision maker. This basic idea has been suggested many times. Algorithm of playing Tic Tac Toe has been presented and tested that works in efficient way. Overall the system works without any bugs.We have built a Tic-Tac-Toe using the C programming language. Tic Tac Toe is a game that is played by people of all ages. It has been developed using the C programming language. Any purpose-driven decision maker can have intelligence as a trait. A Tic Tac Toe algorithm has been presented in the system and tested, and it has proven to be effective. Chiefly, the system is free of errors.

7.REFERENCE

1. Oxford English Dictionary entries for "Noughts and Crosses", "Tick-Tack" and "Tick-Tack-Toe", dictionary.oed.com
2. Jump up to:[a] [b] Wolf, Mark J. P. (16 August 2012). Encyclopedia of Video Games: The Culture,Technology,and Art of Gaming. Greenwood Publishing Group.
3. https://github.com/topics/tic-tac-toe?l=c%2B%2B