

ITC PROJECT

TIC -TAC-TOE- GAME



NAME-SANIDHYA BREJA

ROLL NO-13

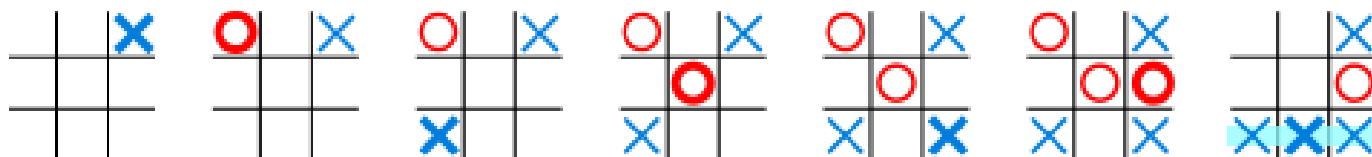
PRN NO-2014110674

INTRODUCTION:

Tic-tac-toe (American English), noughts and crosses (Commonwealth English), or Xs and Os (Irish English) is a paper-and-pencil game for two players who take turns marking the spaces in a three-by-three grid with *X* or *O*. The player who succeeds in placing three of their marks in a horizontal, vertical, or diagonal row is the winner. It is a solved game, with a forced draw assuming best play from both players.

GAMEPLAY-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.

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

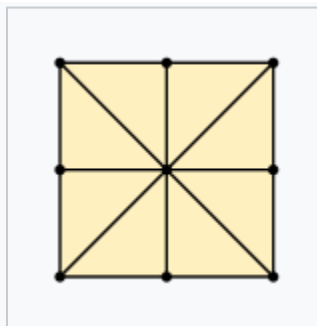


There is no universally-agreed rule as to who plays first, but in this article the convention that *X* plays first is used.

Players soon discover that the best play from both parties leads to a draw. Hence, tic-tac-toe is often

played by young children who may not have discovered the optimal strategy.

Because of the simplicity of tic-tac-toe, it is often used as a pedagogical tool for teaching the concepts of good sportsmanship and the branch of artificial intelligence that deals with the searching of game **tree**. It is straightforward to write a computer program to play tic-tac-toe perfectly or to enumerate the 765 essentially different positions (the state space complexity) or the 26,830 possible games up to rotations and reflections (the game tree complexity) on this space.^[1] If played optimally by both players, the game always ends in a draw, making tic-tac-toe a **futile game**.^[2]



Incidence structure for tic-tac-toe

The game can be generalized to an m,n,k -game, in which two players alternate placing stones of their own color on an m -by- n board with the goal of getting k of their own color in a row. Tic-tac-toe is the 3,3,3-game. Harary's generalized tic-tac-toe is an even broader generalization of tic-tac-toe. It can also be generalized as an n^d game, specifically one in which n equals 3 and

d equals 2. It can be generalized even further by playing on an arbitrary incidence structure, where rows are lines and cells are points. Tic-tac-toe's incidence structure consists of nine points, three horizontal lines, three vertical lines, and two diagonal lines, with each line consisting of at least three points.

STRATEGY-

A player can play a perfect game of tic-tac-toe (to win or at least draw) if, each time it is their turn to play, they choose the first available move from the following list, as used in Newell and Simon's 1972 tic-tac-toe program.

1. **Win:** If the player has two in a row, they can place a third to get three in a row.
2. **Block:** If the opponent has two in a row, the player must play the third themselves to block the opponent.
3. **Fork:** Cause a scenario where the player has two ways to win (two non-blocked lines of 2).
4. **Blocking an opponent's fork:** If there is only one possible fork for the opponent, the player should block it. Otherwise, the player should block all forks in any way that simultaneously allows them to make two in a row. Otherwise, the player should make a two in a row to force the opponent into defending, as long as it does

not result in them producing a fork. For example, if "X" has two opposite corners and "O" has the center, "O" must not play a corner move to win. (Playing a corner move in this scenario produces a fork for "X" to win.)

5. Center: A player marks the center. (If it is the first move of the game, playing a corner move gives the second player more opportunities to make a mistake and may therefore be the better choice; however, it makes no difference between perfect players.)
6. Opposite corner: If the opponent is in the corner, the player plays the opposite corner.
7. Empty corner: The player plays in a corner square.
8. Empty side: The player plays in a middle square on any of the four sides.

The first player, who shall be designated "X", has three possible strategically distinct positions to mark during the first turn. Superficially, it might seem that there are nine possible positions, corresponding to the nine squares in the grid. However, by rotating the board, we will find that, in the first turn, every corner mark is strategically equivalent to every other corner mark. The same is true of every edge (side middle) mark. From a strategical point of view, there are therefore only three possible first marks: corner, edge, or center. Player X

can win or force a draw from any of these starting marks; however, playing the corner gives the opponent the smallest choice of squares which must be played to avoid losing. This might suggest that the corner is the best opening move for X, however another study shows that if the players are not perfect, an opening move in the center is best for X.

The second player, who shall be designated "O", must respond to X's opening mark in such a way as to avoid the forced win. Player O must always respond to a corner opening with a center mark, and to a center opening with a corner mark. An edge opening must be answered either with a center mark, a corner mark next to the X, or an edge mark opposite the X. Any other responses will allow X to force the win. Once the opening is completed, O's task is to follow the above list of priorities in order to force the draw, or else to gain a win if X makes a weak play.

More detailed, to guarantee a draw, O should adopt the following strategies:

- If X plays corner opening move, O should take center, and then an edge, forcing X to block in the next move. This will stop any forks from happening. When both X and O are perfect players and X chooses to start by marking a corner, O takes the center, and X takes the corner opposite the original. In that case, O is free to

choose any edge as its second move. However, if X is not a perfect player and has played a corner and then an edge, O should not play the opposite edge as its second move, because then X is not forced to block in the next move and can fork.

- If X plays edge opening move, O should take center or one of the corners adjacent to X, and then follow the above list of priorities, mainly paying attention to block forks.
- If X plays center opening move, O should take corner, and then follow the above list of priorities, mainly paying attention to block forks.

When X plays corner first, and O is not a perfect player, the following may happen:

- If O responds with a center mark (best move for them), a perfect X player will take the corner opposite the original. Then O should play an edge. However, if O plays a corner as its second move, a perfect X player will mark the remaining corner, blocking O's 3-in-a-row and making their own fork.
- If O responds with a corner mark, X is guaranteed to win, by simply taking any of the other two corners and then the last, a fork. (since when X takes the third corner, O can only take the position between the two X's. Then X can take the only remaining corner to win)

- If O responds with an edge mark, X is guaranteed to win, by taking center, then O can only take the corner opposite the corner which X plays first. Finally, X can take a corner to create a fork and then X will win on the next move.

EXAMPLE-

Consider a board with the nine positions numbered as follows:

1	2	3
4	5	6
7	8	9

When X plays 1 as their opening move, then O should take 5. Then X takes 9 (in this situation, O should not take 3 or 7, O should take 2, 4, 6 or 8):

- $X1 \rightarrow O5 \rightarrow X9 \rightarrow O2 \rightarrow X8 \rightarrow O7 \rightarrow X3 \rightarrow O6 \rightarrow X4$, this game will be a draw.

or 6 (in this situation, O should not take 4 or 7, O should take 2, 3, 8 or 9. In fact, taking 9 is the best move, since

a non-perfect player X may take 4, then O can take 7 to win).

- **X1 → O5 → X6 → O2 → X8**, then O should not take 3, or X can take 7 to win, and O should not take 4, or X can take 9 to win, O should take 7 or 9.
 - **X1 → O5 → X6 → O2 → X8 → O7 → X3 → O9 → X4**, this game will be a draw.
 - **X1 → O5 → X6 → O2 → X8 → O9 → X4 (7) → O7 (4) → X3**, this game will be a draw.
- **X1 → O5 → X6 → O3 → X7 → O4 → X8 (9) → O9 (8) → X2**, this game will be a draw.
- **X1 → O5 → X6 → O8 → X2 → O3 → X7 → O4 → X9**, this game will be a draw.
- **X1 → O5 → X6 → O9**, then X should not take 4, or O can take 7 to win, X should take 2, 3, 7 or 8.
 - **X1 → O5 → X6 → O9 → X2 → O3 → X7 → O4 → X8**, this game will be a draw.
 - **X1 → O5 → X6 → O9 → X3 → O2 → X8 → O4 (7) → X7 (4)**, this game will be a draw.
 - **X1 → O5 → X6 → O9 → X7 → O4 → X2 (3) → O3 (2) → X8**, this game will be a draw.
 - **X1 → O5 → X6 → O9 → X8 → O2 (3, 4, 7) → X4/7 (4/7, 2/3, 2/3) → O7/4 (7/4, 3/2, 3/2) → X3 (2, 7, 4)**, this game will be a draw.

In both of these situations (X takes 9 or 6 as second move), X has a

1
/
3

property to win.

If X is not a perfect player, X may take 2 or 3 as second move. Then this game will be a draw, X cannot win.

- $X1 \rightarrow O5 \rightarrow X2 \rightarrow O3 \rightarrow X7 \rightarrow O4 \rightarrow X6 \rightarrow O8 (9) \rightarrow X9 (8)$, this game will be a draw.
- $X1 \rightarrow O5 \rightarrow X3 \rightarrow O2 \rightarrow X8 \rightarrow O4 (6) \rightarrow X6 (4) \rightarrow O9 (7) \rightarrow X7 (9)$, this game will be a draw.

If X plays 1 opening move, and O is not a perfect player, the following may happen:

Although O takes the only good position (5) as first move, but O takes a bad position as second move:

- $X1 \rightarrow O5 \rightarrow X9 \rightarrow O3 \rightarrow X7$, then X can take 4 or 8 to win.
- $X1 \rightarrow O5 \rightarrow X6 \rightarrow O4 \rightarrow X3$, then X can take 7 or 9 to win.
- $X1 \rightarrow O5 \rightarrow X6 \rightarrow O7 \rightarrow X3$, then X can take 2 or 9 to win.

Although O takes good positions as the first two moves, but O takes a bad position as third move:

- $X1 \rightarrow O5 \rightarrow X6 \rightarrow O2 \rightarrow X8 \rightarrow O3 \rightarrow X7$, then X can take 4 or 9 to win.
- $X1 \rightarrow O5 \rightarrow X6 \rightarrow O2 \rightarrow X8 \rightarrow O4 \rightarrow X9$, then X can take 3 or 7 to win.

O takes a bad position as first move (except of 5, all other positions are bad):

- **X1 → O3 → X7 → O4 → X9, then X can take 5 or 8 to win.**
- **X1 → O9 → X3 → O2 → X7, then X can take 4 or 5 to win.**
- **X1 → O2 → X5 → O9 → X7, then X can take 3 or 4 to win.**
- **X1 → O6 → X5 → O9 → X3, then X can take 2 or 7 to win.**

Tic Tac Toe Game Development using C:

While making a Tic Tac Toe game using C language, it is important to make use of arrays. The Xs and Os are kept in different arrays, and they are passed between several functions in the code to keep track of how the game goes. With the code here you can play the game choosing either X or O against the computer.

This Tic Tac Toe C game is such that you will have to input a numerical character, from 1 to 9, to select a position for X or O into the space you want. For example: if you are playing with O and you input 2, the O will go to the first row – second column. If you want to place O in third row – first column, you have to enter 7. And, it is similar for the other positions.

This has been done this way because it is just a console application without graphics designed in C language. The gotoxy function has been used to print text in any part of the screen.

Did you know?

In the Tic Tac Toe game, there are 765 states of space complexities or over 25,000 possible games on those different positions.

Function Used:

I have divided this project into many functions, and below is a list of those functions. I have only described the gotoxy function in detail. Just go through the source code once, and other functions used are simple and easy to understand.

void menu() – In this mini project, this function displays the menu or welcome screen of this project. Scroll down to view the photo of the menu. With this function, you can select whether you wish to play the game with X or with O.

void go(int n)

void start_game()

void check_draw()

void draw_board()


void player_first()

void put_X_O(char ch, int pos) – This function puts one of the numerical character you input into the respective position in Tic-Tac-Toe. For example: if you are playing with X and you input 2, the X will go to first row – second column. If you want to place X in third row – first column, you have to enter 7. And, it is similar for the other positions.

void gotoxy (int x, int y) – You need to understand this function as it is one of the

most important one used in Tic Tac Toe in C. This function allows you to print text in any place of the screen. Using this function in Code::Blocks requires coding, but it can be directly used in Turbo C.

GAME OUTPUT SCREENSHOTS-



```
C:\Users\sanid\Downloads\Mini Project in C Tic Tac Toe Game\tic tac toe.exe

-----MENU-----
1 : Play with X
2 : Play with O
3 : Exit
Enter your choice:>
```

C:\Users\sanid\Downloads\Mini Project in C Tic Tac Toe Game\tic tac toe.exe

```
-----MENU-----  
1 : Play with X  
2 : Play with O  
3 : Exit  
Enter your choice:>1
```

```
  |  |  |  
--|  |  |--  
  |  |  |  
--|  |  |--  
  |  |  |
```

Your Turn :>

C:\Users\sanid\Downloads\Mini Project in C Tic Tac Toe Game\tic tac toe.exe

```
-----MENU-----  
1 : Play with X  
2 : Play with O  
3 : Exit  
Enter your choice:>1
```

```
  X |  |  |  
--|  |  |--  
    |  O |  |  
--|  |  |--  
    |  |  |
```

Your Turn :> 1

C:\Users\sanid\Downloads\Mini Project in C Tic Tac Toe Game\tic tac toe.exe

```
-----MENU-----  
1 : Play with X  
2 : Play with O  
3 : Exit  
Enter your choice:>1
```

```
  x  |  x  |  o  
-----  
      |  o  |  
-----  
      |    |
```

Your Turn :> 2

C:\Users\sanid\Downloads\Mini Project in C Tic Tac Toe Game\tic tac toe.exe

```
-----MENU-----  
1 : Play with X  
2 : Play with O  
3 : Exit  
Enter your choice:>1
```

```
  x  |  x  |  o  
-----  
  x  |  o  |  
-----  
  o  |    |
```

Your Turn :> 4

Computer wins

