# Unbeatable Computer Algorithm for a 3x3 Tic-Tac-Toe board

This algorithm was developed entirely from scratch and makes use of some common patterns found by me after extensive research. Things to note: the player always starts first; this document will refer to the computer as 'O' and the player as 'X', unless told otherwise; and the document does not cover the sub-algorithm 'block/win'. The patterns can be categorized into three cases: case 1, when the player doesn't occupy the middle square upon their first turn; case 2, when the computer does occupy the middle square upon its first turn; and case 3, when the player occupies two squares diagonally adjacent to each other. Generally, occupying the center square is the best first move, and this gives a 50% chance of winning the game. This document will delve deeper into the cases listed above, providing detailed explanations and examples for each one.


Case 1: the player doesn't occupy the center square upon their first move

The computer should always aim to occupy the middle square, giving it control over the board. After doing so, the computer should then proceed to occupy the middle square in any row or column that is empty; in other words, a row or column with no squares occupied by the user.
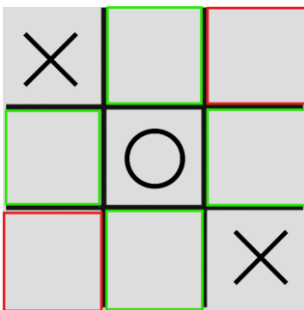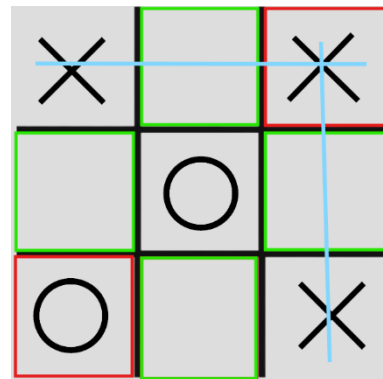

On the example on the left, the green boxes denote the possible 'safe' zones for the computer to occupy (places that won't result in a loss), and the red boxes indicate the 'dangerous' zones for the computer to occupy, places that can result in a loss.
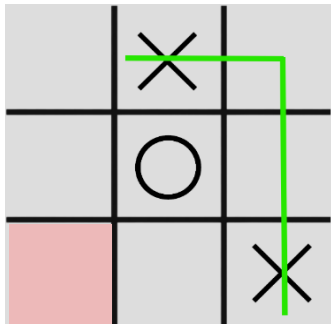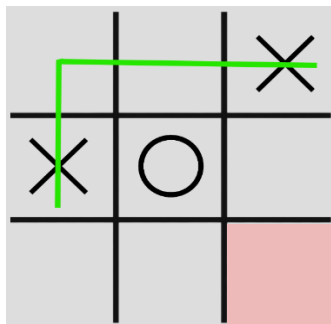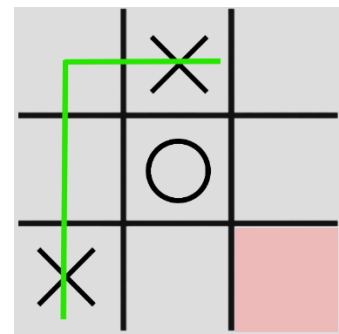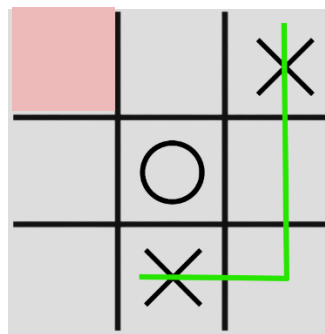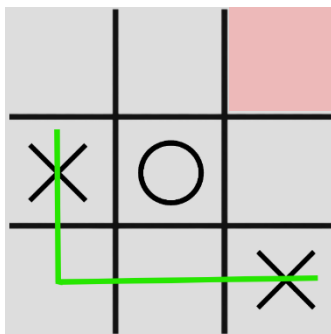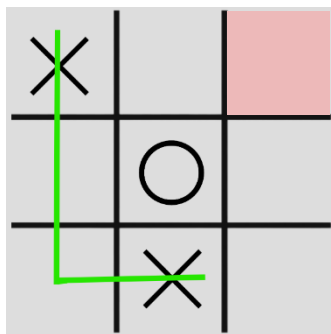
The computer opts for the dangerous zone:
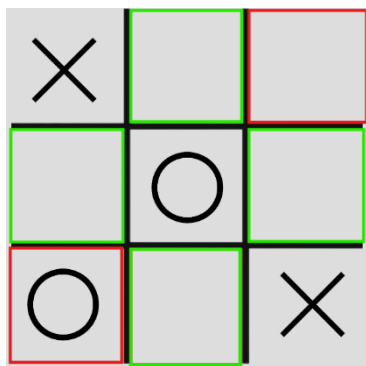
Here, the blue lines represent potential winning states for the player.

If the computer opts for the safe zones, it will ensure that the game is at least going to result in a draw if not a win.

The L-shaped pattern:

These are all possible examples of the L-shaped pattern on a 3 by 3 tic-tac-toe board.

The green lines highlight the pattern.

The red square represents the 'danger' zone

When such a pattern is made, the computer should always try to occupy a square inside that L-shaped pattern, preferably a middle square in that row.

Case 2: the player occupies the center square upon his or her first move
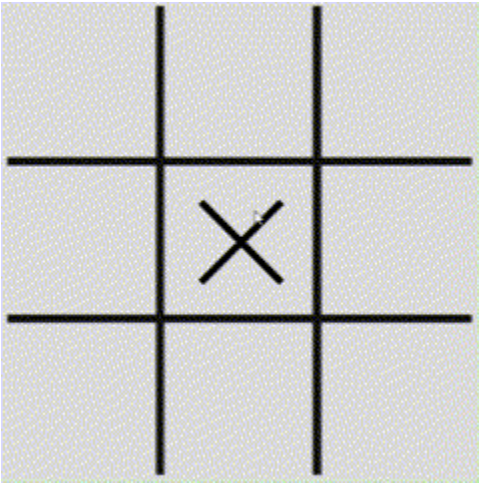
It is important that the computer tries to occupy a corner square, given that the player has already occupied the middle square.

The example on the left shows the result of the computer failing to meet the conditions mentioned above. The condition is true for all 3 by 3 tic-tac-toe boards since the same patterns can be reflected on the board, regardless of orientation.

Case 3: the user forms a pattern a 'side by side' pattern
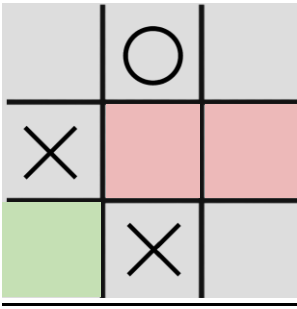
When the player decides to form the pattern shown of the left, the computer should always aim to cover the square in between the pattern, denoted by the green square. As such, this prevents the player from a two-way win.

The computer opts for the dangerous zone:
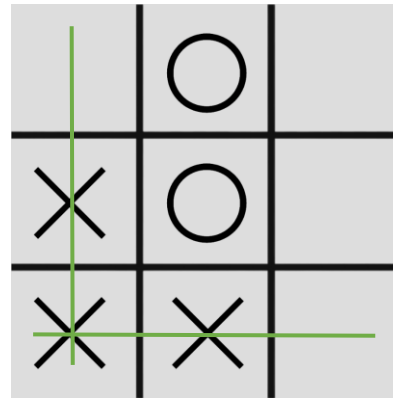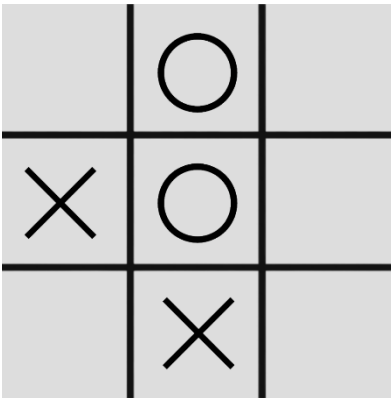
As seen above, it is a guaranteed win for the player as there is a two-way win on the next move, as depicted by the green arrows.