'Can't Lose' Algorithm

Below is a basic algorithm for the computer to not lose a game. You need to use this algorithm for two things in your Minor Project task:

1. Use the following algorithm for the **hard setting of your computer opponent in the Tic-Tac-Toe task**. Note that this is an algorithm for "not losing" as opposed to winning. (i.e if you are a good player it should always end up being a draw.)
2. Use this algorithm to create a **flowchart in your documentation**. This flowchart needs to have the correct symbols and flows logically.

Nought's and Crosses Game Tactic Algorithm

These are the tactics that can be used to ensure the player does not lose. Note that this does not mean that the player will win, in most cases these tactics will lead to a draw.

First Player Tactics (O)

- Move 1: Go in a corner.
- Move 2: If the other player did not go there then go in the opposite corner to move 1. Otherwise go in a free corner.
- Move 3: If there are two Xs and a space in a line (in any order) then go in that space. Otherwise if there are two Os and a space in a line then go in that space. Otherwise go in a free corner.
- Move 4: If there are two Xs and a space in a line (in any order) then go in that space. Otherwise if there are two Os and a space in a line then go in that space. Otherwise go in a free corner.
- Move 5: Go in the free space.

Second Player Tactics (X)

- Move 1: If Player 1 went in the centre then go in one of the corners otherwise go in the centre.
- Move 2: If there are two O's and a space in a line (in any order) then go in that space. Otherwise go to an Adjacent Square
- Move 3: If there are two O's and a space in a line (in any order) then go in that space. Otherwise if there are two X's and a space in a line then go in that space. Otherwise go in a free square.