**Lab17: Understanding Other's Code**

I was assigned to check the code of Nadim SAAD, so the Tic-Tac-Toe game.

According to the presented code. I understand the logic/workflow of the game as:

**Logic of Tic-Tac-Toe game presennted in the code:**

* Create a dictionary with 9 elements which will be holding the values presenetd on the board game.
* Create a function which will allow to dispaly a pseudo board game.
* Then they defined a function called playersturn in which in fact the whole logic of the game is constructed. So in this function they start by:
  + Saying that player 'X' starts
  + Then they set the counter, which keeps track of number of marks enterd into the boardgame
  + The maximum number of turns is 9 that is why they created a loop in order to visualise 9 times the board and explain which turn it is.
  + In the next step they ask for the players input 'move'
  + Then they check with the if statement if the place in the board is empty. If it is empty then the symbol of a player associated to the variable turn is put into the gaming\_board and the counter increases by one.
  + In the next step they check if one of the players won. This is done by the long if statement which is activated after 5 moves/turns, which is logic because the win is possible after inserting 5 symbols into the board game.
  + If one of the conditions of wining the game is fulfiled (8 variants; the same 3 symbols in vertical lines - 3 sases; the same 3 symbols in horizontal lines - 3 cases, and 2 cases of 3 the same symbols in diagonal lines), then the message is displayed saying that the game is over, and a message which indicates the winer.
  + Then we exit this long if statement and we check if the counter reached 9. If it did and there is no win that means that we reached a tie.
  + Finally after exiting another nested if statement we come back to the initial if and we ask the player to select a free space.
  + Then the game is changing the turn of the player, so X truns to O or O turns to X.
  + Then the code is asking for another user input. If the players want to play again or not.
  + Finally, the fuction calls itself. I guess this is the problem of identation, and the reason why it is impossible to exit a game. I am assuming that this was a mistake.
* At the very end of the code the function with the game logic is called.

**My comments to the proposed code:**

I think that there are right siolutions, but too much is done in one fuction. I would create smaller functions in which I can check if thhe player won, if the board game is full, which means tie and a function to display board (in fact I will reuse the function proposed). I would also change keys in the dictionary from strings to numbers. Then I would use those functions to create the logic of the game. Below is the logic of the tic tac toe game proposed by me.

**Potential imrovements**

1. Create better user experience (welcome messages, rules)
2. Simplify code (while loops instead of elif / split the bug function into small ones)