Data structures and Algorithms Coursework

# Introduction

The task that I have been given is creating text based Tic-Tac-Toe game by implementing several of data structures and algorithms in the C programming language. The game must have a game board, players, pieces and positions. The game must record history of play and enable earlier games to be automatically replayed from this record. The game must support undo where once a move is made, you can undo the move to the game state before where players can undo moves to initial game state. The game should also have a redo feature that allows redo moves which been UN done. The features that the Tic TAC Toe game has a register and login where users can register an account by setting a username and password. The password is encrypted then the encrypted and username is written to a file. The login feature is where user can login into the game as long the username and password equal to username and password in a certain line of a file. The other features are a multiplayer game where two humans can play with each other by entering the square number that a user want to position their mark on until a player wins or there is a draw. The game moves is then replayed from initial state to end state in a way where each position is placed every time a user presses a key until finished state. A player can then decide to play again or go back to the menu. There is also two other types of games where a player can play against a computer. One game is where computer makes a random move every time it is turn. The other version is where a minimax implementation is used to allow a computer to see all of the possible moves and chooses the best move every time the computer makes a move. There is two replays search system. One is where a user can enter a game id key and a certain moves from games that have been played would be replayed subsequently. Another is where a user can enter a game id key and certain moves from a game from among the games that certain user have played in the past.

# Design

### Tic TAC Toe Game – Multiplayer Design

I have used a 1D array of chars for the game board where each char represent a square of the board which initially would have ‘1’, ‘2’, ‘3’, ‘4’, ‘5’,’6’,’7’, ‘8’, ‘9’. Depending on what square number, that a player have positioned a piece on the game board, the string would be replaced with a certain piece, either a nought(‘O’) or cross(‘X’), depend on what player have positioned a move. I have a used a 1D array because it enables random access of elements which makes accessing elements by position quicker every time a player makes a certain move. It also have better cache locality compare to data structure like linked lists, hash map, stacks and queues which improves performance of the game. I have used a 1D array instead of a 2D array for game board because the total bytes of a 1D array just depend on size of char \* size of array where as the total bytes of a 2D array just depend on (size of char) \* size of first index \* size of 2nd index which mean 1D array takes less memory space compare to a 2D array. Also, the 1D array can be received in pointer, sized array or a unsized array where as the received parameter must define rightmost dimension of a array which makes the game board easier to be implemented in a 1D array compare to being implemented in a 2D array.