Tic Tac Toe Report

INTRODUCTION

This report presents overview of design, implementation and evaluation of Tic Tac Toe console game. Application is written entirely in C programming language.

Core features are:

- 2 players New Game mode
- Undo and Redo during the game
- Replay of any previous game

DESIGN

Where it comes to visual representation user is presented with choice to start a new game or replay one of previously played games. In the appendices there is screenshot showing initial screen with gameboard (Figure 1)

From technical point of view my main concern was to structure my code properly, with such big application I divided it into several files to improve readability and separate sections of code that are not related to each other.

All entities are represented by structs and eventually arrays of them.

Structures can be divided into **main ones**, used in New Game as well as Replay mode:

- Player, containing information about player name and mark
- Gameboard, holding its size and mark on each field

and **specifically used for game replay**:

- Single_game_info, that contain information about place in file, where specific moves can be found, as well as both players details.
    - Dynamically allocated array of Single_game_info keeps general information of all games played in the past.
- Single_move_details, that contain information about type of action performed by user (move, undo, redo) and field affected.
    - Dynamically allocated array of Single_move_details keeps general information of all actions performed during single game

Additionally, to support “undo” and “redo” I implemented stack where field numbers are held. During new game all key information, and moves are saved into txt file in data directory.

Enhancements

If I had more time I would add functionality to play with computer, as well as allow players to play games on bigger boards. Other interesting feature that I would add is possibility to play on torus board, and 3 sides of 3d cube.

Critical Evaluation

I am particularly proud of algorithm to check whether someone won a match, it is capable to determine whether someone won match on bigger board of n x n size. On the other hand, I am certain that changing the way of storing game history would improve reading from file speed, as currently history is stored in txt file, not binary.

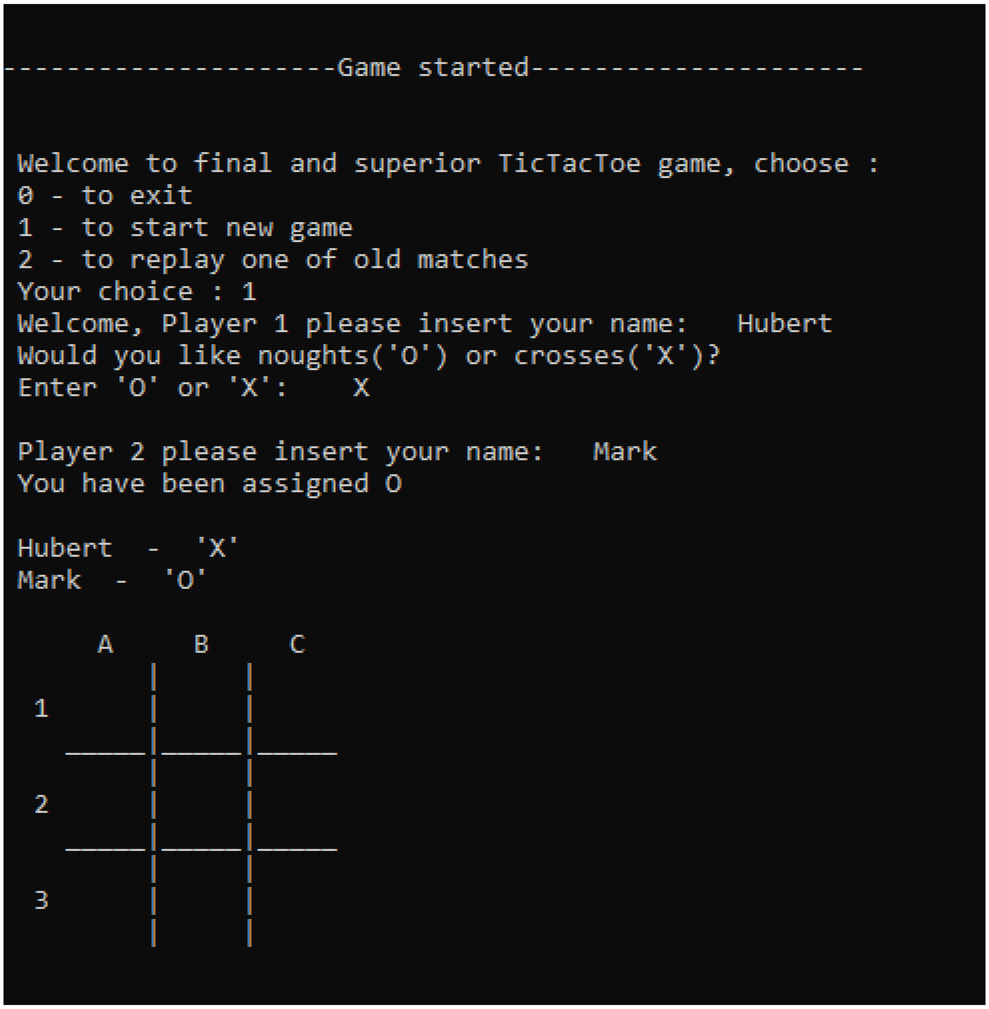
Personal Evaluation

During this project I learned a lot about how data is managed by an operating system, and how particular structures can easy manipulating it. Biggest challenge to me was to implement variety of features already available in higher level languages and dynamically allocate memory.

References

https://www.tutorialspoint.com/cprogramming/

APPENDICES



```
---------------------Game started---------------------


Welcome to final and superior TicTacToe game, choose :
0 - to exit
1 - to start new game
2 - to replay one of old matches
Your choice : 1
Welcome, Player 1 please insert your name:   Hubert
Would you like noughts('O') or crosses('X')?
Enter 'O' or 'X':    X

Player 2 please insert your name:   Mark
You have been assigned O

Hubert  -  'X'
Mark  -  'O'

     A     B     C
        |     |
 1      |     |
   _____|_____|_____
        |     |
 2      |     |
   _____|_____|_____
        |     |
 3      |     |
        |     |
```

Figure 1