# Coursework – Report

## Alexander Koenig, [40322645@live.napier.ac.uk](mailto:40322645@live.napier.ac.uk)

### 27/03/2019

## Introduction

The task of the coursework was to implement a version of “Tic Tac Toe” using the C programming language, using the knowledge gained from the lectures and labs – the core goal should be to understand and use various data algorithms and structures. In my implementation, only the game itself has been implemented – no other features.

## Design

A char array holds the full field, each field is represented by one character. These characters change based on which player enters and what field the player wants to change. If a user enters a wrong character, or an already taken field, the turn is forfeited. After every turn, there is a check whether someone won, or if there is a tie.

## Enhancements

If I had more time, I would add the turn-based system with undo and redo, I would also add a way to play more than one game, and maybe record the results as well, potentially in an external file to keep track even when the program is closed.

## Critical Evaluation

The game works quite well on a very basic level, mostly because it is such a simple game. The algorithms used seem to fit in well, though no memory-based testing was done. In general, the game is missing a lot of components to call it complete, however.

## Personal Evaluation

I’m not going to sugar-coat my performance here: It was abysmal. I had planned to start this coursework months ago, and here I am, two hours before hand-in. While the personal problems and current state of mind about attending University certainly played a factor, they should not be taken into consideration for grading. I know I could have done this much better, I know I am fit to sit – but I didn’t do it, and I should pay the price for it.

In general, though, I am surprised that even with so little time left, I produced a working game. I did not really have many challenges, though there was one thing that took me a while to understand: Every other turn, the “scanf” function would just be skipped. Thanks to Stackoverflow, however, this was quickly understood and resolved.

## References

* Stackoverflow (<https://stackoverflow.com>) – Various pages, especially useful:
  + scanf Getting Skipped [duplicate] (<https://stackoverflow.com/questions/14484431/scanf-getting-skipped>)