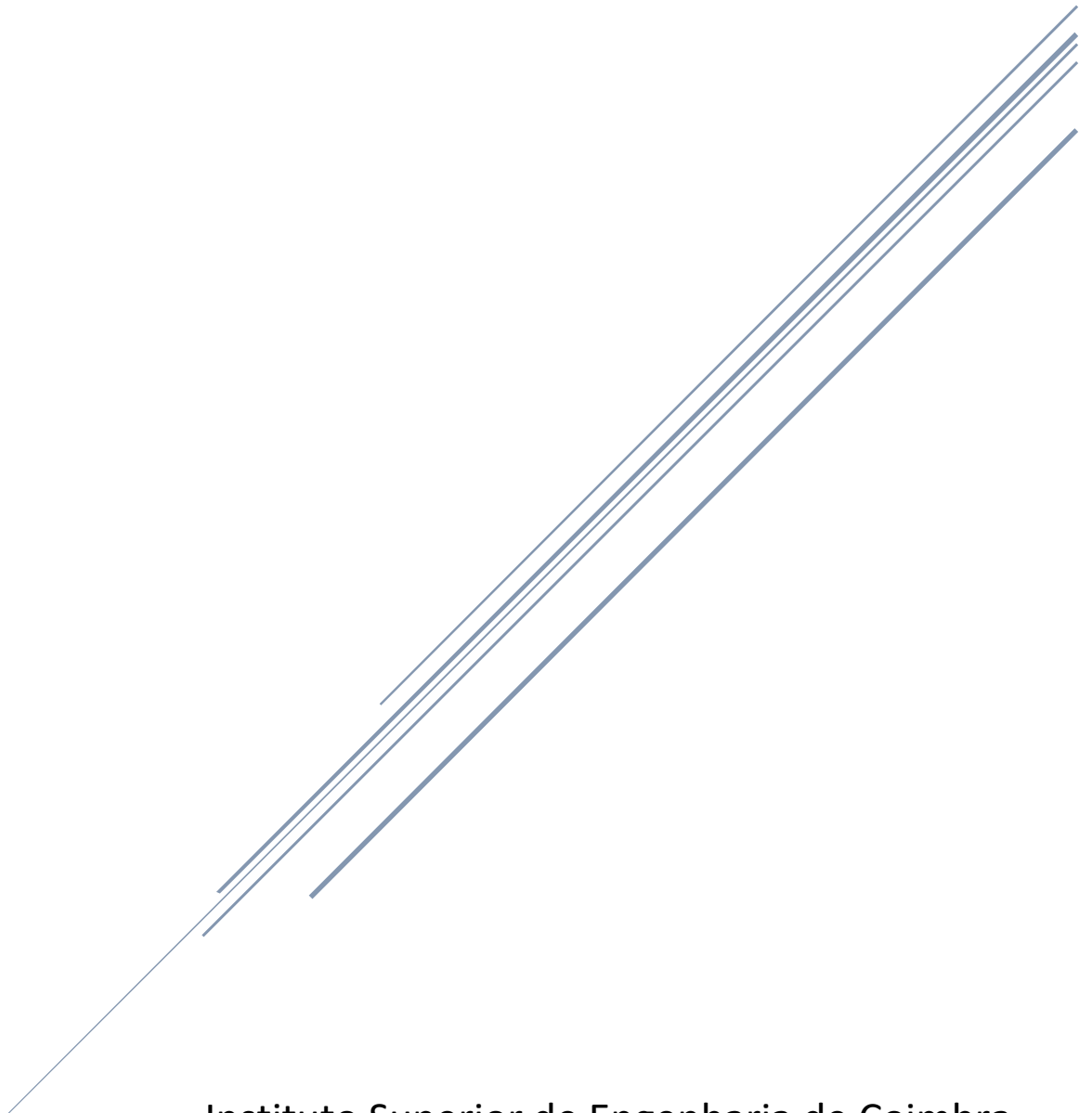


# PROGRAMAÇÃO – 2021/2022

Trabalho Prático



Instituto Superior de Engenharia de Coimbra  
Licenciatura Engenharia Informática(CE)

## Índice

Conclusões	4
Decisões de implementação	3
Estruturas dinâmicas	3
Funcionalidades Implementadas	4
Identificação do ambiente de trabalho	3
Organização do projeto	2

## 1. Organização do projeto

O projeto tem a seguinte organização de ficheiros e principais funcionalidades:

- **Jogo**

Interface inicial que permite iniciar um jogo novo, carregar um jogo anterior e seleccionar o modo de jogo.

- Interface inicial com o utilizador;
- Escolher o modo de jogo;
- Iniciar novo jogo ou carregar um jogo antigo;
- Inicializar o tabuleiro;

- **SuperTic**

Estrutura que suporta toda a lógica e informação do jogo.

- Ponteiro para o histórico de jogadas (lista ligada de estruturas histórico);
- Alocação dinâmica do tabuleiro;
- Funções para inicialização, impressão do tabuleiro de jogo e interface de texto;
- Funções para carregar e guardar o jogo;
- Funções para a execução de jogadas, quer por computador, utilizador ou jogadas automáticas;
- Funções para a verificação de vitória do tabuleiro inteiro e de jogadas;

- **Tic**

Estrutura que suporta uma matriz.

- Funções para impressão da matriz;
- Funções de verificação de vitória e de espaços disponíveis na matriz;

- **Historico**

Estrutura que funciona com lista ligada, onde cada elemento é uma jogada feita durante o decorrer do jogo, contem também todos os elementos necessários para a escrita/leitura da lista de e para ficheiro.

- Funções de listagem, inserção e eliminação de elementos na lista;

Foram também usados os ficheiros disponibilizados para funções de escolha aleatória, criação e libertação de matriz.

## 2. Identificação do ambiente de trabalho

O projeto foi desenvolvido na totalidade usando o “CLion”.

## 3. Estruturas dinâmicas

Dinamicamente o projeto é suportado pela estrutura **SuperTic**, onde vai conter uma matriz 3x3 do tipo **Tic** onde cada posição tem um ponteiro para uma matriz 3x3, um ponteiro para o início do histórico de jogadas (lista ligada), e variáveis do tipo **Int** para guardar o modo de jogo, jogador atual, estado do jogo e número de jogadas.

O número de jogadas e o modo de jogo vai ser guardados no início do ficheiro para ajudar na leitura e na continuação do jogo no modo de jogo correto, e depois é guardado o histórico de jogadas.

## 4. Decisões de implementação

Ter uma estrutura que engloba todos os elementos necessários aligeira a implementação do programa e de futuras implementações, como foi o caso do histórico que bastou incorporar um ponteiro para a estrutura no SuperTic, e implementar a funcionalidades necessários nos ficheiros referentes ao Historico, que depois vão ser chamadas facilmente nas funções do SuperTic.

A partir do ponto onde tinha a estrutura bem definida e relativamente solido, o tempo de desenvolvimento das funcionalidades essenciais para a logica de jogo aumento muito, mas até la chegar demorou um bom par de horas frustrantes.

Penso que tendo a estrutura desta maneira, fica mais acessível adicionar funcionalidades e modificar por exemplo para um tabuleiro 4x4, tendo apenas de alterar as verificações, validações de jogadas e pouco mais.

Tentei minimizar a complexidade do algoritmo separando-o em várias funções para facilitar a sua legibilidade e reutilizar o código ao máximo, tornando assim mais fácil encontrar e resolver eventuais problemas no programa.

Em termos de regras de jogo, tentei dar um bom número de opções, entre escolher em que quadrado começa, ou ser uma escolha aleatória, o que permite mais variabilidade no início do jogo. O utilizador também tem muito controlo sobre o que deseja ver e fazer antes de cada jogada.

Em casos onde se jogue para uma posição onde o tabuleiro para onde vai a seguir esta ganho ou completo, a jogada é contabilizada de seguida esse jogador decide um outro quadrado válido, ou pode optar por ir para um quadrado válido aleatório, assim fica nas mãos dos utilizadores como querem jogar.

## 5. Funcionalidades Implementadas

- **Definição das estruturas de dados:** Realizado;
- **Correção das funcionalidades implementadas:** Realizado;
- **Manipulação de estruturas dinâmicas:** Realizado;
- **Manipulação de ficheiros:** Realizado, parcialmente;
- **Simplicidade/funcionalidade da interface com o utilizador:** Realizado.

## 6. Conclusões

Durante a implementação deste trabalho, aprendi verdadeiramente o valor que as estruturas trazem para a programação, bem como manipulação de listas ligadas e as diferenças de manuseamento de ficheiros binários e de texto. Esses foram os aspetos mais difíceis durante o trabalho, muitas vezes foi necessário dar um passo atrás para dar dois em frente.

Embora seja um jogo fácil de entender, á medida que a complexidade aumenta, tentar manter o programa o mais possível isento de erros, torna-se muito difícil, e é muito fácil perder horas em torno de pequenos erros que desvendam outros erros ainda maiores ao longo do código.

Usando o melhor das minhas capacidades e com limite de tempo devido ao sobre carregamento de outros trabalhos, penso que entrego um bom projeto, embora ainda existam alguns erros, que ainda não apareceram, mas provavelmente vão aparecer na defesa certamente.

Com a conclusão deste trabalho sinto-me muito mais apto para desenvolver trabalhos com mais de umas dezenas de linhas de código em c, sinto-me mais confiante em lidar com estruturas, listas ligadas e ficheiros.