



FACULDADE DE  
CIÊNCIAS E TECNOLOGIA  
UNIVERSIDADE D  
**COIMBRA**

# **Relatório do miniprojeto do Jogo da Força**

**Gonçalo Bastos – 2020238997**

**José Nogueira – 2020233342**

**25 – 04 – 2022**

Definimos para o nosso jogo 2 classes: a classe *Jogo* que tem todos os atributos e métodos relacionados com o funcionamento do jogo ou manipulados pelo mesmo; e a classe *Jogador* derivada da classe *Jogo* onde adicionamos alguns métodos constantes que serão uteis. Definimos ainda algumas funções globais usadas em todo o programa. Todas estas definições foram colocadas num header, chamado *jogo\_forca.h*.

Na nossa *main* apenas chamamos 2 funções, uma para printar a capa do jogo, e uma para o menu.

A função *menu* e todas as outras funções globais estão declaradas num ficheiro .cpp chamado *globalf.cpp*. Neste ficheiro também declaramos os métodos que não têm a ver com os modos de jogo, esses métodos estão num ficheiro .cpp chamado *modos.cpp*.

Na função global *menu()*, a partir da qual tudo funciona, definimos a nossa estrutura de dados, que será uma tabela de Jogadores alocada na memória dinâmica, definida a partir da classe *Jogador*. Por segurança definimos um número máximo de 100 jogadores. Todos os dados e informação relevante para o funcionamento do jogo estarão nessa tabela. Antes de tudo, ainda passamos a informação relevante guardada no backup para a memória (dinâmica) do programa, ou seja, para essa tabela. Este backup é um ficheiro .txt que vai conter toda a informação relevante sobre jogadores, e pontuações acumuladas, bem como qualquer outra informação manipulada pelo programa. Desta forma não haverá perda de dados.

Depois, como é óbvio, apresentamos o menu ao user e é-lhe dada a possibilidade de:

- **Iniciar um novo jogo.** Onde ele pode escolher:

1- continuar a jogar no mesmo save onde já jogou introduzindo o seu nome;

2- continuar a jogar um jogo que deixou suspenso

(Para funcionameto de 1 e 2 tomamos como opção o programa fazer uma pesquisa sequencial pela memória para encontrar o jogador, pois a informação não precisa de estar ordenada);

3- começar como sendo um novo jogador.

- **Escolher o modo antes de jogar**, no caso de ser um novo jogador. Se não for, vai-lhe ser perguntado mais adiante na opção 1 do menu se pretende alterar o modo. Obs: se escolher o modo básico ou médio será pedido também a dificuldade.

- **Mostrar o ScoreBoard.** Onde aparecem as pontuações dos jogadores ordenadas. Escolhemos fazer esta ordenação por inserção e a sua implementação está no método *ordenalInsercaoPontuacao()*.

- **Simplemente Sair.** Se sair, fazemos a libertação da memória. O guardar da informação no backup é feito ao longo do programa, e também neste momento de forma assegurar que não se perdem dados.