

5089 - Desafio 2 - Corrida ATEC

Ao desenvolver este programa tenha em atenção o bom uso dos comentários, de indentação e o correto nome das variáveis!!!! Bom trabalho!

Enquadramento

A ATEC realiza anualmente uma corrida de solidariedade. Por forma a informatizar as consultas de resultados, a ATEC solicitou à turma a realização de um pequeno programa que carregue os resultados gerais e permita a consulta, através do número de dorsal, das informações de corrida de cada atleta que terminou a prova

Ficheiro de Resultados Gerais

Considere um ficheiro de dados de texto que contém os resultados de uma prova de atletismo. Cada linha do ficheiro representa um participante e contém a seguinte informação:

- Posição: Posição geral em que acabou a prova. Exemplo de ficheiro no servidor GIT
- Escalão: Caracter seguido de um número. O caracter pode ser M (masculino) ou F (feminino) e o número corresponde à idade mínima do escalão.
- Dorsal: Número do dorsal. Atribuído durante a inscrição
- Nome: Nome do participante.
- Sexo: M (masculino) ou F (feminino)
- Tempo: Tempo que demorou a completar a prova.

Cada atleta é uma linha do ficheiro e os campos do ficheiro encontram-se delimitados por um tab (\t).

Exemplo linhas de um ficheiro (Repare que a primeira linha do ficheiro não deve ser carregada para o programa):

```
Pos.      Escalão Pos. Esc.  Dorsal  Nome      Sexo      Tempo
1      M35 1    6    Jose Moreira    M    1:07:49 AM
2      M35 2    8    Rui Pedro Silva M    1:08:32 AM
3      M35 3    7    Nicodemus Biwott M    1:08:44 AM
4      M20 1   696 Bruno Miguel Ramos Da Silva M    1:08:48 AM
5      M20 2   14    Rui Muga      M    1:08:54 AM
6      M20 3   13    Mihail Lalev   M    1:08:55 AM
7      M20 4   672 João Antunes    M    1:11:42 AM
8      M20 5    5    Paulo Mendes    M    1:11:57 AM
9      M35 4   11    Oscar Mendes    M    1:13:02 AM
10     M20 6   900 Bruno Carvalho  M    1:13:13 AM
...
...
```

Trabalho a efetuar

Ler Ficheiro e criar lista encadeada de atletas

- O nome do ficheiro deve ser passado ao programa por um argumento de linha de comando:
`./corrida results-atec.txt`
 - Caso o ficheiro não exista (erro ao abrir em modo de leitura), o programa deve terminar com a seguinte mensagem de erro **ERRO! Ficheiro não encontrado**;
- Crie estrutura(s) que permitam guardar a informação e gerar uma lista encadeada (simplesmente encadeada ou duplamente encadeada é uma opção sua);
- Utilize alocação dinâmica de memória para inserir na Lista cada uma das linhas, ou seja cada um dos atletas;
- Todo o ficheiro deve ser carregado numa lista encadeada;

Comportamento esperado do programa

- O Programa ao iniciar, e caso não haja erro na abertura do ficheiro e criação da lista deve apresentar a mensagem **Insira N° dorsal->**;
- O utilizador deve inserir o número do dorsal e uma das mensagens deve ser mostrada:
 - **Caso o número inserido seja negativo (<0):** **ERRO! Número de dorsal inválido**
 - **Caso o número inserido seja 0:** **A sair...**
 - **Caso o número inserido seja >0 mas atleta não exista:** **ERRO! Atleta não existe**
 - **Caso o número inserido seja >0 e atleta exista:** Escreve a informação do atleta conforme o exemplo de funcionamento apresentado abaixo:
- O programa só termina a sua execução quando o utilizador inserir **0**;
- A memória alocada dinamicamente deve ser libertada antes de sair.

Exemplos de funcionamento do programa quando não foi passado o ficheiro ou este não exista:

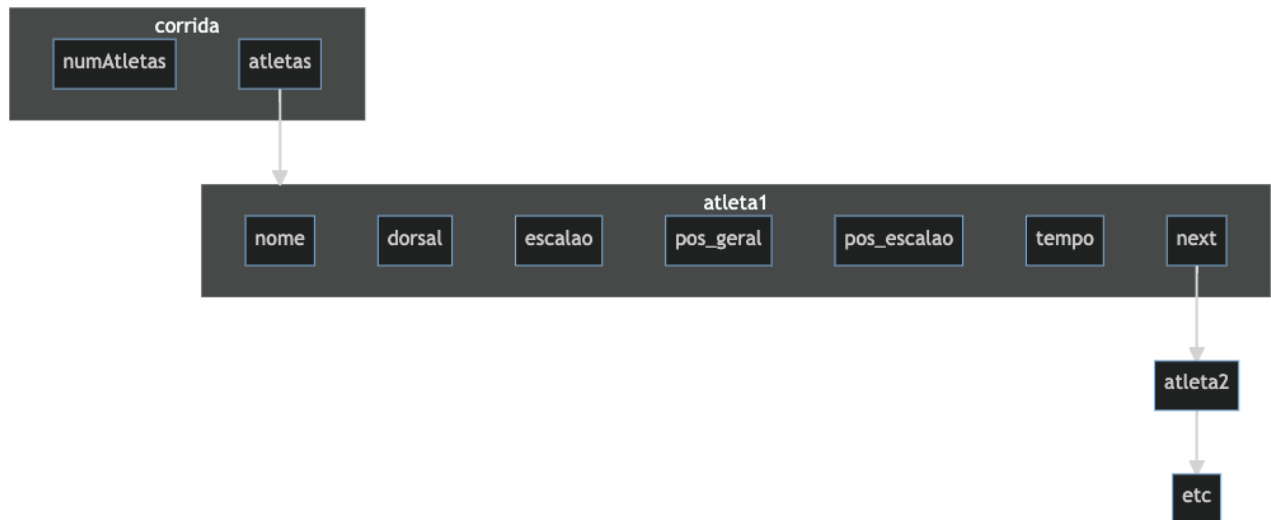
```
ERRO! Ficheiro não encontrado
A sair ...
```

Exemplos de funcionamento do programa quando o ficheiro é carregado com sucesso

```
Insira N° dorsal->777
ERRO! Atleta não existe
Insira N° dorsal->-3
ERRO! Número de dorsal inválido
Insira N° dorsal->900
-----
Atleta: Bruno Carvalho
Sexo: M
Escalão: M20
Pos. Geral: 10
Pos. Escalão: 6
Tempo Prova: 1:13:13
-----
Insira N° dorsal->0
A sair ...
```

Sugestões

- Teste o programa com o valgrind, para aferir a correta manipulação de memória. Programas sem erros na manipulação de memória são valorizados
- Considere, na leitura das linhas do ficheiro que a dimensão máxima da linha é 1024 caracteres
- Utilize a função strtok para partir a string pelos tabs (\t) por forma a extrair os campos. veja exemplo em [tutorialspoint](#) ou em [geeksforgeeks](#) ou em Português no site [Wiki C](#), ou ainda um video no [Youtube](#)
- Os campos das estruturas (por exemplo o nome) só devem ocupar o espaço estritamente necessário (Necessário fazer malloc)
- Sugere-se a seguinte estrutura na criação das structs



- Organize o seu código criando uma biblioteca. por exemplo corrida.h e corrida.c
 - No ficheiro main.c deixe apenas a função main e outras necessárias para ler o ficheiro e analisar os argumentos de linha de comando
- Comente o seu código. É considerado na avaliação

Entrega

A entrega deve ser feita de dois modos:

- Upload para o Git (obrigatório criar uma pasta 5089-desafio2)
- Resposta à tarefa através do upload de um printscreen com parte do código e consola do aluno (obrigatório)