Laboratório 11

Objectivos do laboratório: Pretende-se que o aluno exercite a pesquisa e ordenação.

Exercício 1

Considere uma associação (e.g., clube de futebol). Os dados dos associados encontram-se armazenados num ficheiro, associados.txt, disponível em anexo, e com o seguinte formato:

```
<Nome> <apelido> <numero de associado> <dia nascimento> <mês> <ano>
```

Considere que a lista esta ordenada pelo numero de associado.

Construa um programa que satisfaça os seguintes requisitos:

- Defina o tipo **TipoAssociado** que permita armazenar os dados de cada associado.
- Defina na função principal o vetor **v_associados** com MAX_ELEM elementos de variáveis do tipo TipoAssociacao.
- Crie a função **ler_dados** que leia o ficheiro associados.txt e preencha a estrutura devidamente.
- Crie a função **pesquisar** que:
  - Peça ao utilizador para inserir um número de associado.
  - Faça uma pesquisa sequencial, por forma a indicar ao utilizador o nome desse associado, ou uma mensagem adequada caso não exista na lista.
  - Faça uma pesquisa binária, por forma a indicar ao utilizador o nome desse associado, ou uma mensagem adequada caso não exista na lista.
- Crie a função **fazer_estatisticas**:
  - Indique no ecrã quantos associados existem.
  - Qual o apelido mais comprido.
  - Qual o associado mais velho.
- Crie a função **ordenar** que deverá ordenar o vetor por idades. Deverá usar o algoritmo de ordenação por seleção.
- Guarde num novo ficheiro a listagem ordenada.

Utilize funções por forma a que o main seja pequeno. Guarde a definição da estrutura e as funções num ficheiro biblioteca.h.

Exercício 2

Escreva um programa que satisfaça os seguintes requisitos:

- Defina na função **main** um vetor com 10000 inteiros. O valor de 10000 é especificado recorrendo a uma constante SIZE.
- Crie a função **preenche_primos** que preenche o vetor com todos os números primos existentes entre 1 e 10000. Preencha sequencialmente o vetor. Desta forma o vetor ficará ordenado. Imprima os numeros primos que o programa identificou. A função deverá retornar a quantidade de números primos existentes. De facto, o vetor não ficará totalmente preenchido.
- Peça ao utilizador que insira um número entre 1 e 10000. Caso não respeite a condição volte a perguntar.
- Informe o utilizador se o número é ou não primo. Verifique de duas formas, recorrendo:
    - **pesquisa_sequencial**, que faz a pesquisa sequencial no vetor de primos. A função deverá receber como argumento o vetor, a quantidade de primos e o número a pesquisar. Pare se encontrar um primo, ou se o numero for menor do que o elemento do vetor que está a analisar (numero < vetor[i]), pois como está ordenado, não vale a pena continuar a pesquisar. Imprima uma "*" por cada iteração (primeira instrução dentro do while).
    - **pesquisa_binaria**, que faz a pesquisa binária no vetor de primos. A função deverá receber como argumento o vetor, a quantidade de primos e o número a pesquisar. Imprima uma "*" por cada comparação que for feita na pesquisa sequencial.

  Volte a pedir ao utilizador para inserir um outro número. Caso insira o número 0, termine.
- Experimente com os números 1, 10 e 9999. Compare os tempos dos dois algoritmos de pesquisa.

```
...(por questões de espaço não se imprimem todos os números primos)... 9833 9839 9851 9857
9859 9871 9883 9887 9901 9907 9923 9929 9931 9941 9949 9967 9973

Indique um inteiro entre 1 e 10000 para verificar se e primo: 9999

Pesquisa sequencial:
**********************************************************************************************
**********************************************************************************************
**********************************************************************************************
**********************************************************************************************
**********************************************************************************************
**********************************************************************************************
**********************************************************************************************
**********************************************************************************************
**********************************************************************************************
**********************************************************************************************
**********************************************************************************************
**********************************************************************************************
**********************************************************************************************
**********************************************************************************************
**********************************************************
9999 nao e primo

Pesquisa binaria: ***********
9999 nao e primo
```