**No fim do exame deverá confirmar que submeteu corretamente os ficheiros no SIGEX, com o nome indicado no enunciado.**

**1** [7 valores] Tendo por base as bibliotecas de estruturas de dados apresentadas em Programação 2, implemente as funcionalidades pedidas nas duas alíneas seguintes no ficheiro **prob1.c** fornecido. Sempre que conveniente utilize as funções disponíveis.

**1.1** [4 valores] Implemente a função encontra\_nomes, que coloca num novo vetor apenas os nomes que estão simultaneamente presentes em ambos os vetores de entrada (que não devem ser alterados).

vetor\* encontra\_nomes(vetor \*vec1, vetor \*vec2)

Quando ocorrer um erro o retorno deverá ser NULL.

Indique ainda num comentário no início do código da função qual a complexidade do algoritmo que implementou.

Depois de implementada a função, o programa deverá apresentar:

Lista resultante contem 50 nomes.

**1.2** [3 valores] Implemente a função filtra\_nomes que permite obter uma nova fila com os nomes que comecem pela *string* definida pelo parâmetro str.

fila\* filtra\_nomes(lista \*lst, char \*str)

Adicionalmente, esses nomes devem ser removidos da lista de entrada lst. Quando ocorrer um erro o retorno deverá ser NULL. Sugestão: utilize a função strstr da biblioteca string.h.

Depois de implementada a função, o programa deverá apresentar (para str=”MARIA”):

Nomes filtrados (total de 25):

MARIA ANGELICA WEBER

MARIA AUXILIADORA AZEVEDO GRANJEIRO

MARIA CLARA SANTOS MAYRINK

...

MARIANA SCHNEIDER

**\*\* Submeta o ficheiro prob1.c no SIGEX \*\***

**2** [10 valores] Tendo por base as bibliotecas de estruturas de dados apresentadas em Programação 2, implemente as funcionalidades pedidas nas duas alíneas seguintes no ficheiro **prob2.c** fornecido. Sempre que conveniente utilize as funções disponíveis.

**2.1** [4 valores] Implemente a função grafo\_intersecao que, dados dois grafos dirigidos com o mesmo número de vértices, cria e devolve um novo grafo correspondente à interseção entre os dois. Considere que a interseção entre dois grafos dirigidos ***g1*** e ***g2*** com o mesmo número de vértices é um grafo dirigido que tem também o mesmo número de vértices, mas cujas arestas são apenas as que existem simultaneamente em ***g1*** e ***g2***.

grafo\* grafo\_intersecao(grafo \*g1, grafo \*g2)

Os parâmetros da função são os grafos a partir dos quais se pretende obter a interseção. A função deve retornar o novo grafo se for bem sucedida ou NULL em caso contrário, incluindo erro nos parâmetros. Indique ainda num comentário no início do código da função qual a complexidade do algoritmo que implementou e uma breve justificação (máximo 20 palavras).

Depois de implementada a função, o programa deverá apresentar:

0: 4

1: 0->4->5

2:

3:

4:

5: 4

**2.2** [4 valores] Implemente a função imprime\_ordenado que imprime as chaves de uma tabela de dispersão por ordem decrescente do valor associado a cada chave.

void imprime\_ordenado(tabela\_dispersao \*td)

O parâmetro da função é o apontador para a tabela de dispersão, que deverá ser verificado.

Depois de implementada a função, o programa deverá apresentar:

bay of dragons

essos

the riverlands

the crownlands

...

kingdom of sarnor

the stormlands

**2.2** [2 valores] Indique a complexidade da solução implementada na alínea anterior e uma justificação clara e sucinta (máximo 50 palavras) no comentário assinalado para o problema 2.3 no ficheiro prob2.c. Para complementar a resposta poderá incluir comentários no problema 2.2 indicando a complexidade das respetivas instruções.

**\*\* Submeta o ficheiro prob2.c no SIGEX \*\***