

Nome do estudante: N.º

Informação aos estudantes: A consulta permitida inclui slides das aulas teóricas, livros e outros materiais impressos. Não serão permitidas folhas manuscritas avulsas de qualquer tipo ou acesso à Internet (Tablets, portáteis, etc). Telemóveis deverão permanecer **DESLIGADOS** durante a duração do exame. Responder os seguintes grupos de questões em folhas separadas (uma folha para cada grupo): {1, 2}, {3, 4}, {5, 6}.

1. [5 valores] Pretende-se avaliar quantos caminhos existem desde o início até ao fim de um jardim. Considere que:

- O jardim visto de cima é representado por uma matriz de dimensão $M \times N$, com todas as posições inicializadas a 0;
- A posição inicial encontra-se nas coordenadas (0, 0);
- A posição final está no canto inferior direito da matriz, ou seja, nas coordenadas (M-1, N-1);
- A cada momento só é possível a deslocação para a direita ou para baixo;
- Todas as posições da matriz são possíveis, ou seja podem ser visitadas.

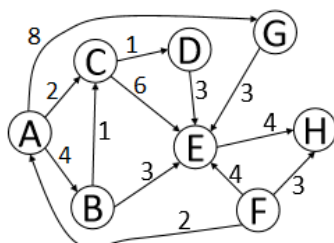
Pretende-se saber quantos caminhos são possíveis para chegar do início ao fim do jardim. Por exemplo, para uma matriz 3x2 a resposta é 3, e existem os seguintes caminhos: {(0, 0), (0, 1), (0, 2), (1, 2)}; {(0, 0), (0, 1), (1, 1), (1, 2)}; {(0, 0), (1, 0), (1, 1), (1, 2)}. Utilizando código C++, resolva as alíneas abaixo:

- [2 valores] Resolva o problema usando uma abordagem recursiva e comente, justificando, a complexidade temporal desta abordagem.
- [2 valores] Resolva o problema usando uma abordagem de Programação Dinâmica, justificando a complexidade temporal desta abordagem. Utilize a matriz original para chegar à solução.
- [1 valor] Considere agora que algumas das posições da matriz podem ter o valor 1, significando que tal posição não pode ser visitada. Altere a implementação anterior para suportar este cenário. Caso não tenha feito a alínea b), altere a implementação da a).

2. [2 valor] Considere uma lista L de números inteiros, ordenada de forma decrescente. Queremos saber se um número N está contido na lista. Apresente, em pseudocódigo, duas abordagens para resolver o problema:

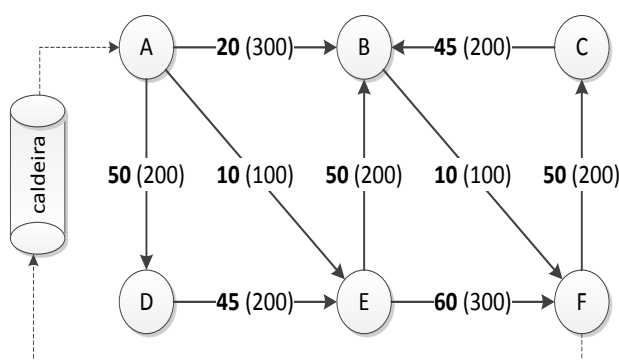
- [1 valor] considere uma solução com complexidade temporal $O(N)$.
- [1 valor] considere uma solução com complexidade temporal $O(\log_2(N))$.

3. [3 valores] O grafo seguinte representa localidades (vértices A, B, ..., H) e respetivas estradas (arestas). As estradas são de sentido único e o valor indicado na figura representa o tempo (médio) que demora a travessia dessa estrada.



- [1 valor] Apresente uma ordenação topológica do grafo. Explique.
 - [0,5 valores] Usando o algoritmo de Dijkstra, determine o caminho mais rápido quando se pretende ir da localidade A para a localidade H. Apresente todos os passos intermédios, explicando.
 - [1 valor] Num grafo G pesado e dirigido, pretende-se determinar o caminho mais curto entre dois vértices V_x (origem) e V_y (destino), mas passando obrigatoriamente pelo vértice V_p (passagem). Apresente um algoritmo, em pseudocódigo ou C++, para a resolução deste problema.
 - [0,5 valor] Sabendo que a passagem numa localidade obriga ao pagamento de uma *taxa*, identifique o caminho que obriga a pagar menos taxas, quando se pretende ir da localidade A para a localidade H. Que tipo de pesquisa deve usar, neste caso?
4. [3 valores] Num laboratório de genética, estudos relacionados com ADN obrigam à transmissão frequente de um elevado volume de dados. Estes dados incluem o uso de 4 caracteres diferentes (A,C,G,T). Num estudo em particular, a informação usada contém os 5 caracteres com a seguinte frequência: A: 2000, C: 4000, G: 2000; T: 4000.
- [1,5 valores] Apresente uma árvore de codificação de Huffman, para a informação do estudo atrás enunciado. Explique.
 - [0,5 valores] De acordo com a árvore obtida na alínea anterior, codifique a seguinte sequência ADN: "AGCCT".
 - [1,0 valores] O algoritmo de Huffman pode originar codificações de diferentes tamanhos para os códigos de cada item, devido ao não determinismo na construção da árvore de codificação (nomeadamente, na junção de sub-árvores). Na transmissão da informação codificada, pretende-se que esta seja efetuada, tanto quanto possível, a uma taxa constante, isto é, pretende-se minimizar as diferenças entre tamanhos dos códigos de cada item: $\min(\sum_i frequencia_i * (tam_i - average_tam)^2)$. Que cuidado deve ter, na construção da árvore de codificação de Huffman, para satisfazer tal requisito? Usando o cenário acima indicado, ilustre a sua explicação.

5. [3 valores] Uma caldeira é utilizada para aquecer os ambientes de uma habitação, a partir de radiadores a ela ligados e localizados nos vários compartimentos da casa. O grafo da figura mostra o circuito concebido para o sistema de aquecimento, sendo alimentado pela caldeira a partir do vértice *A*, sendo a água reenviada à caldeira a partir do vértice *F*, para evitar desperdícios. As arestas representam os radiadores localizados nos vários compartimentos, com os valores das respectivas capacidades de fluxo de água, indicados em **bold**, e as potências de aquecimento produzidas pelos radiadores, indicadas entre parênteses. Responda às alíneas seguintes, justificando a sua resposta:



- [0,5 valores] Sabendo que a caldeira instalada é capaz de trabalhar a um volume máximo de 75 UVs (unidades de volume), esta caldeira é apropriada para a habitação representada pelo grafo?
 - [1,5 valor] Considerando que o custo com calefação é proporcional à potência produzida pelos radiadores como função do volume de água utilizado, qual a potência total de calor que produzirá a caldeira?
 - [0,5 valores] Verifique se o circuito foi original concebido correctamente, ou seja, se todos os radiadores estão a receber água da caldeira, ou se haverá algum compartimento que não esteja a ser aquecido.
 - [0,5 valores] Verifique se a caldeira será capaz de gerar mais calor na habitação, caso esteja ligada ao vértice *D*, em adição ao vértice *A*. Quanto será o volume total de água no sistema?
6. [4 valores] Dado um grafo não dirigido $G=(V, E)$, um subconjunto $W \subseteq V$ é dito ser um **clique** do grafo G se, e se somente, $\forall u, v \in W$ então $(u, v) \in E$. Ou seja, um **clique** de um grafo não dirigido é um subconjunto dos seus vértices tal que, para quaisquer pares de vértices u e v neste subconjunto, existe uma aresta do grafo que liga os vértices u e v . Demonstre que o problema associado a encontrar um **clique** num grafo não dirigido é NP-completo, explicando, passo-a-passo, o procedimento utilizado.

Sugestão: Utilize o problema de isomorfismo de subgrafos (*Subgraph Isomorphism, SI*) na sua solução, sabendo-se que *SI* é um problema NP-completo. Mais formalmente, dados dois grafos G_1 e G_2 , *SI* procura definir uma função unívoca, f , que mapeia vértices de G_1 a vértices de G_2 , de tal forma que, se existe uma aresta (u, v) em G_1 , então existirá uma aresta $(f(u), f(v))$ pertencente a G_2 .

Bom Exame!