

Estruturas de Dados e Algoritmos

Exame de Época Normal

José Jasnau Caeiro

21 de junho de 2022

LER ATENTAMENTE

Submeta a resposta desta questão, nos próximos **30 minutos**. Submeta no local *moodle* do IPBeja¹ usando o seu acesso de aluno.

Entregue só um ficheiro, usando a linguagem de programação **Julia**, designado por **main.jl** que engloba todas as suas respostas e contendo todo o código necessário para demonstrar que tem uma resposta correta.

Se utilizar *packages* externos indique sob a forma de comentário qual o *package*. Refira todas as fontes consultadas sob a forma de comentário na sua resposta.

No caso de não referir as fontes consultadas será considerada tentativa de fraude. E se existirem respostas com estrutura de código similar, de 2 ou mais alunos, também será considerada fraude e anulado todo o exame.

Identifique claramente:

- o seu nome completo;
- o seu número de aluno;
- a alínea a que respondeu².

Leia com calma a questão e tente não depender excessivamente da informação na *Internet*.

Questão 4

Crie uma pasta para a questão designada por **edaq4**.

- 1.a) (2 valores) Instale os pacotes para programação com grafos para a linguagem de programação **Julia** designados por **Graphs.jl** e **SimpleWeightedGraphs.jl**³.

Programe uma função designada por **fgXXXXX**, em que **XXXXX** é o seu número de aluno, que:

- utiliza o construtor **SimpleWeigthedGraph(n)** para representar o grafo da Figura 1;

¹<http://cms.ipbeja.pt>

²Caso não o faça pode ser anulada a resposta.

³A documentação do pacote encontra-se em <https://github.com/JuliaGraphs/SimpleWeightedGraphs.jl>.

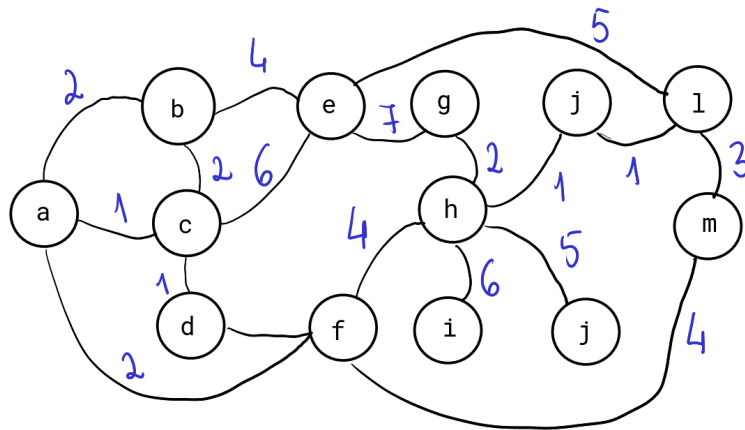


Figura 1: Grafo não ponderado.

- tem como entrada uma tabela de tópos;
 - cada tópo representa um ramo e tem 3 campos: (s, t, w) em s é o nó de origem, t é o nó destino e w é a ponderação associada ao ramo;
 - devolve o grafo resultante.
- 1.b) (3 valores) Programe o algoritmo de pesquisa em largura tendo como origem o nó e , e aplique ao grafo da alínea anterior para obter a árvore resultante da pesquisa em largura. programe uma função designada por `imprimir_bfs()` que envia para a consola a árvore de pesquisa em largura. Esta função deve devolver a profundidade máxima da árvore.