

AULA 13 – O TIPO ABSTRATO DE DADOS GRAPH – PARTE II

***** Para os dois guiões (12 e 13) sobre grafos, entregue num ficheiro ZIP todos os ficheiros de implementação em que completou código *****

Na aula teórica foram apresentados alguns algoritmos para, dado um vértice inicial, determinar a **árvore dos caminhos mais curtos com origem nesse vértice inicial**.

Neste guião deverão ser implementados e testados os seguintes algoritmos que constroem essa árvore dos caminhos mais curtos:

- Algoritmo usando uma **QUEUE** para referenciar os vértices candidatos, cujo rótulo foi recentemente alterado.
- Algoritmo de **Dijkstra**, usando uma **PRIORITY QUEUE** para obter o próximo vértice a juntar à árvore dos caminhos mais curtos.
- Algoritmo de **Bellman-Ford**, na sua **versão melhorada**, em que se interrompe o varrimento do conjunto de arestas do grafo ao ser detetado que **não houve qualquer alteração de rótulos** de vértices no varrimento anterior.

TAREFAS

- **Completar e testar** os três módulos que implementam os algoritmos pedidos.
- Para um mesmo grafo, e para diferentes vértices iniciais, **comparar** as árvores de caminhos mais curtos obtidas. Em que circunstâncias é que essas árvores podem ser diferentes?
- Executar os algoritmos para os vários grafos fornecidos em ficheiro.

Atenção:

Os vértices de um grafo estão sequencialmente numerados: 0, 1, 2, ...

Deve respeitar os protótipos das funções definidos nos vários ficheiros cabeçalho.

Pode criar funções auxiliares (static) locais a cada módulo.