Instituto Politécnico de Leiria

Escola Superior de Tecnologia e Gestão

Matemática Discreta - Componente PL

EI (D+PL)

Ano letivo 2018/2019 - 2.º Sem. Ficha prática 7

Algoritmo de Warshall e Algoritmo do caminho mais curto

Seja G um grafo orientado com m vértices, $v_1, v_2, ..., v_m$, e A a sua matriz de adjacência.

Quando o número de vértices é elevado, a matriz de adjacência A poderá ser fornecida através de um ficheiro Excel. Para ler a informação contida neste tipo de ficheiro, podem usar-se os seguintes comandos:

Comandos para ler informação de um ficheiro Excel	
`A=readxls('localização do ficheiro')` `B=A(k)`	Lê o ficheiro e guarda a informação que ele contém na variável `A` Guarda em `B` o que estiver na k-ésima folha de `A`
`C=B.value`.	Converte em valores numéricos os dados que estão em `B` e guarda-os na variável `C`

Apesar de ser possível determinar a matriz de caminhos P do grafo G através da soma das potências da matriz de adjacência A, fazendo

$$P = \texttt{bool2s}\left(A + A^2 + A^3 + ... + A^m\right)$$

sendo `bool2s` o comando que efetua a substituição de qualquer entrada não nula de uma matriz por 1, este método envolve muitos cálculos.

O algoritmo de Warshall, que é apresentado de seguida, permite obter a matriz de caminhos P do grafo G de uma forma mais eficiente.

Algoritmo de Warshall (para determinar a matriz de caminhos P)

OBJECTIVO: Determinar a matriz de caminhos P do grafo G.

Seja A a matriz de adjacências do grafo G com m vértices.

Neste algoritmo são construídas as matrizes $P_0, P_1, ..., P_m$, de ordem m; em cada iterada é construída a matriz P_k, sendo cada entrada desta matriz dada por

$$P_k\left[i,j\right] = \begin{cases} 1, & \text{se existe um caminho simples de } v_i \text{ para } v_j, \text{ que não usa outro vértice} \\ & \text{exceto, possivelmente, } v_1, v_2, ..., v_k \\ 0, & \text{caso contrário} \end{cases}.$$

Assim:

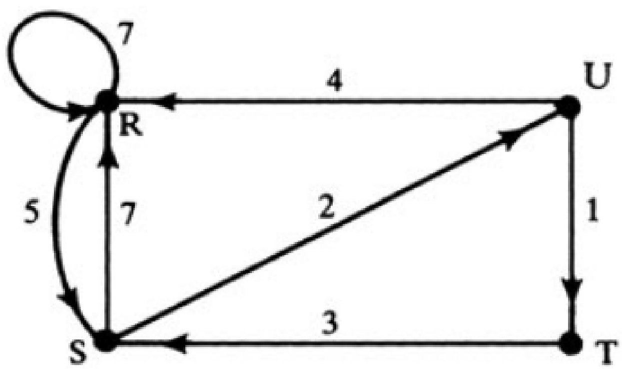
- na iterada $k = 0$ temos $P_0 = A$;
- para $1 \leq k \leq m$ temos $P_k[i,j] = P_{k-1}[i,j] \vee \left(P_{k-1}\left[i,k\right] \wedge P_{k-1}\left[k,j\right]\right)$, onde $1 \leq i,j \leq m$.

Grafos ponderados

Definição 1 *Suponha que G é um grafo ponderado, ou seja, que a cada aresta e de G é associado um valor não negativo $w(e)$, a que chamamos* ***peso*** *(ou* ***custo****) de e. Podemos representar o grafo G à custa da sua* ***matriz de pesos****, $W = (w_{ij})$, que é definida da seguinte forma:*

$$w_{ij} = \begin{cases} w(e)\,, & \text{se existe uma aresta } e \text{ de } v_i \text{ para } v_j \\ \\ 0, & \text{caso contrário} \end{cases}.$$

Exemplo 1 *Considere o grafo G cujo conjunto de arestas é dado por $V = \{R, S, T, U\}$:*



A matriz de pesos de G é a seguinte:

$$W = \begin{bmatrix} 7 & 5 & 0 & 0 \\ 7 & 0 & 0 & 2 \\ 0 & 3 & 0 & 0 \\ 4 & 0 & 1 & 0 \end{bmatrix}.$$

Definição 2 *O* ***peso (custo) de um caminho*** *é calculado somando os pesos de todas as arestas que o formam.*

No grafo do exemplo 1, podemos observar que:

- o peso do caminho TSR é igual a 10, já que resulta da soma do peso do caminho TS, 3, mais o peso do caminho SR, 7.

- existe uma aresta do vértice S para o vértice R com peso igual a 7; no entanto, é possível encontrar um outro caminho entre estes dois vértices, SUR, de peso inferior (igual a 6).

- apesar de não existir uma aresta do vértice U para o vértice S, existe um caminho entre eles, URS, com peso igual a 9; no entanto, é possível encontrar um outro caminho entre estes dois vértices de peso ainda menor, UTS, de peso igual a 4.

Algoritmo do caminho mais curto (ou Algoritmo do caminho de peso mínimo)

OBJECTIVO: Determinar a matriz Q, designada por **matriz de pesos mínimos**, que indica o peso do caminho de menor peso entre os vértices i e j, ou seja, $Q = (q_{ij})$ tal que

$$q_{ij} = \text{peso do caminho de menor peso entre o vértice } i \text{ e o vértice } j.$$

Seja W a matriz de pesos do grafo G com m vértices.

Neste algoritmo são construídas as matrizes $Q_0, Q_1, ..., Q_m$, de ordem m; em cada iterada é construída a matriz Q_k, sendo que:

- na iterada $k = 0$ temos: se $W\left[i,j\right] \neq 0$ então $Q_0\left[i,j\right] = W\left[i,j\right]$; se $W\left[i,j\right] = 0$ então $Q_0\left[i,j\right] = \infty$;
- para $1 \leq k \leq m$ temos: $Q_k[i,j] = \min\left\{Q_{k-1}[i,j] \quad , \quad Q_{k-1}\left[i,k\right] + Q_{k-1}\left[k,j\right]\right\}$, onde $1 \leq i,j \leq m$.

Associada à matriz Q deverá ser também construída uma matriz M, designada por **matriz dos caminhos de peso mínimo,** em que a entrada m_{ij} indica qual é o caminho de menor peso entre o vértice i e o vértice j, cujo peso é igual a q_{ij}, o valor que se encontra na linha i e coluna j da matriz Q.

Exemplo 2

A aplicação do algoritmo do caminho de peso mínimo ao grafo ponderado G, apresentado anteriormente, permite obter a seguinte sequência de matrizes, sendo Q_4 a matriz de pesos mínimos e M_4 a respetiva matriz dos caminhos de peso mínimo:

$$Q_0 = \begin{pmatrix} 7 & 5 & \infty & \infty \\ 7 & \infty & \infty & 2 \\ \infty & 3 & \infty & \infty \\ 4 & \infty & 1 & \infty \end{pmatrix} \qquad M_0 = \begin{pmatrix} RR & RS & - & - \\ SR & - & - & SU \\ - & TS & - & - \\ UR & - & UT & - \end{pmatrix}$$

$$Q_1 = \begin{pmatrix} & & & \\ & & & \\ & & & \\ & & & \end{pmatrix} \qquad M_1 = \begin{pmatrix} & & & \\ & & & \\ & & & \\ & & & \end{pmatrix}$$

$$Q_2 = \begin{pmatrix} & & & \\ & & & \\ & & & \\ & & & \end{pmatrix} \qquad M_2 = \begin{pmatrix} & & & \\ & & & \\ & & & \\ & & & \end{pmatrix}$$

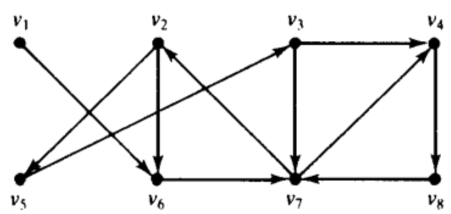
$$Q_3 = \begin{pmatrix} & & & \\ & & & \\ & & & \\ & & & \end{pmatrix} \qquad M_3 = \begin{pmatrix} & & & \\ & & & \\ & & & \\ & & & \end{pmatrix}$$

$$Q_4 = \begin{pmatrix} 7 & 5 & 8 & 7 \\ 6 & 6 & 3 & 2 \\ 9 & 3 & 6 & 5 \\ 4 & 4 & 1 & 6 \end{pmatrix} \qquad M_4 = \begin{pmatrix} RR & RS & RSUT & RSU \\ SUR & SUTS & SUT & SU \\ TSUR & TS & TSUT & TSU \\ UR & UTS & UT & UTSU \end{pmatrix}$$

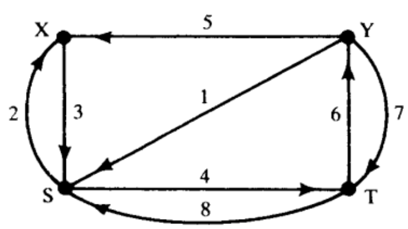
Exercícios propostos

1. Dada a matriz de adjacências A do grafo G, construa uma função com o nome "`Warshall`(A)" que forneça a matriz de caminhos de G. Aplique esta função aos seguintes grafos para determinar a respetiva matriz de caminhos:

   (a)



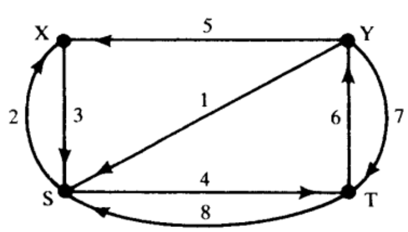
   (b)



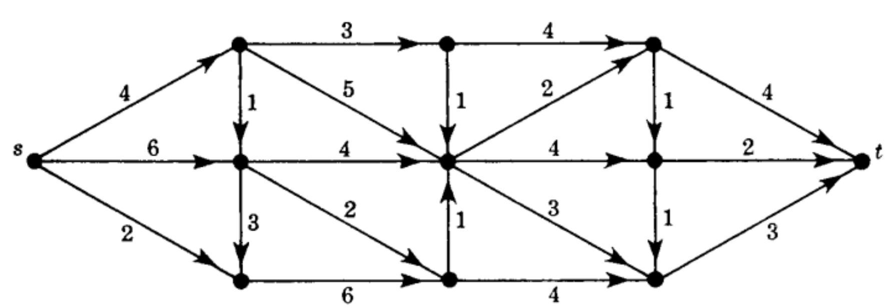
2. Dada a matriz W de pesos de um grafo orientado ponderado G, construa uma função com o nome "`Warshall_MIN`(W)" que forneça a matriz de pesos mínimos, Q, e a matriz dos caminhos de peso mínimo, M, do grafo G.

3. Aplique a função construída no exercício anterior aos seguintes grafos:

   (a)

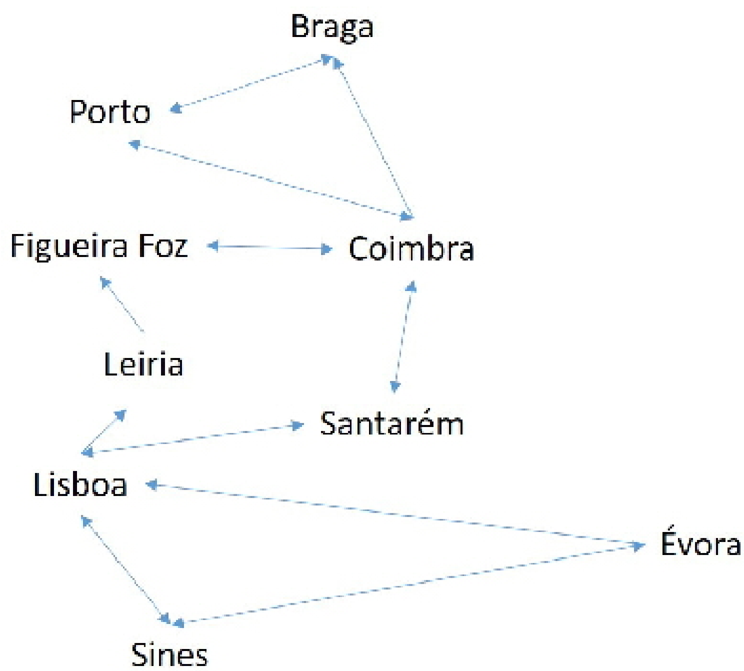


   (b)



   (c) Relativamente ao grafo da alínea anterior, determine o caminho mínimo entre os vértices s e t.

4. Uma empresa de Caminhos de Ferro está a operar em Portugal. No seguinte grafo pode visualizar os trajetos que esta empresa oferece:



No ficheiro "grafos_f7.xls", disponível no Moodle, encontra a matriz de adjacências do grafo correspondente, uma matriz com as distâncias entre as cidades, uma matriz com os custos e a tabela das identificações.

(a) Carregue para o Scilab a matriz de adjacências, a matriz com a duração e a matriz com os custos da viagem, relativas ao grafo dado. As cidades têm a seguinte identificação:

Braga	Porto	Coimbra	F. Foz	Leiria	Santarém	Lisboa	Évora	Sines	
↓	↓	↓	↓	↓	↓	↓	↓	↓	
1	2	3	4	5	6	7	8	9	→ COLUNA

(b) Construa a função `viagem_custo` tal que o comando `viagem_custo(matriz_custo,partida,chegada)` devolva ao utilizador a viagem com o custo mínimo e que, adicionalmente, indique também a duração da viagem e o seu percurso.

(c) Construa a função `viagem_duracao` tal que o comando `viagem_duracao(matriz_duracao,partida, chegada)` devolva ao utilizador a viagem de duração mínimo e que, adicionalmente, indique também o custo da viagem e o seu percurso.