



Segunda Prova — 10 de Julho de 2015

■ QUESTÃO 1 (2,5 pt)

A estrutura *union-find* pode ser utilizada para detecção de ciclos em grafos simples (não-dirigidos, sem auto-arestas) através do seguinte algoritmo:

“Dado um grafo $G = (V = \{v_1, \dots, v_n\}, E)$, comece com uma partição de n conjuntos, cada um contendo exatamente um vértice. Em seguida, para cada aresta $(v_i, v_j) \in E$, se v_i e v_j não pertencem ao mesmo conjunto, então una-os e continue; senão pare indicando que G contém um ciclo.”

Considere então a aplicação do algoritmo acima sobre o grafo G dado pelas listas de adjacências

$$\begin{array}{lll} A \rightarrow C, E & E \rightarrow A, B & I \rightarrow J \\ B \rightarrow D, E, F & F \rightarrow B & J \rightarrow H, I, K, L \\ C \rightarrow A, G & G \rightarrow C, K & K \rightarrow G, J \\ D \rightarrow B, L & H \rightarrow J & L \rightarrow D, J \end{array}$$

supondo que as arestas são processadas na ordem acima $(A, C), (A, E), (B, D), (B, E), (B, F), (C, G), \dots$ e que são utilizadas as heurísticas de *união ponderada* e *compressão de caminhos*.

Represente a estrutura *union-find* ao final da execução do algoritmo e indique se o mesmo acusou ou não a presença de ciclos.

■ QUESTÃO 2 (2,5 pt)

Um percurso em largura (BFS) também pode ser utilizado para detecção de ciclos em grafos simples. Para isso, atribuímos a cada vértice uma cor correspondente a seu estado durante a BFS: a cor branca (B) corresponde a vértices ainda não visitados; a cor cinza (C) corresponde a vértices que começaram a ser visitados e cujos vizinhos ainda estão sendo visitados; e a cor preta (P) corresponde a vértices completamente visitados. Assim, todos os vértices são inicialmente brancos. Um vértice só é enfileirado se estiver branco. Imediatamente antes de ser enfileirado, o vértice é pintado em cinza. Após todos os vizinhos (brancos) de um vértice terem sido enfileirados, ele é marcado como preto. Um ciclo é detectado sempre que um vértice sendo visitado tiver um vizinho cinza.

Considere a execução dessa versão do percurso em largura para detecção de ciclos sobre o grafo da Questão 1. Complete a tabela a seguir correspondente à execução desse percurso

Ordem	Fila	Cores	Visitados
0	(A)	(C, B, B)	()
1	(C, E)	(P, B, C, B, C, B, B, B, B, B, B, B)	(A)
⋮	⋮	⋮	⋮

■ QUESTÃO 3 (2,5pt)

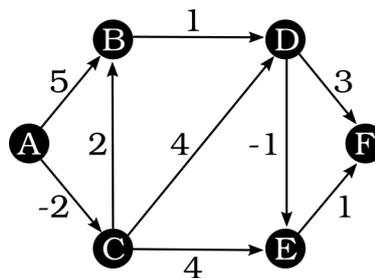
Considere o diagrama a seguir correspondente a uma execução do Algoritmo Dijkstra.

	A	B	C	D	E	F
0	0	∞	∞	∞	∞	∞
1	0	15	∞	30	20	∞
2	0	15	55	25	20	∞
3	0	15	45	25	20	35
4	0	15	45	25	20	35
5	0	15	40	25	20	35

- a) Represente o grafo dirigido com 6 vértices e 8 arestas correspondente a essa execução do algoritmo.
- b) Represente também o vetor de precursores correspondentes a essa execução.

■ QUESTÃO 4 (2,5 pts)

Considere o grafo



Complete a matriz de programação dinâmica a seguir, correspondente à execução do Algoritmo Bellman-Ford no grafo acima para caminhos mínimos a partir do vértice A. (Para cada linha k e vértice t , a entrada correspondente da matriz contém o valor de $\delta^k(s = A, t)$)

k	A	B	C	D	E	F
0	0	∞	∞	∞	∞	∞
1	:	:	:	:	:	:

