Teoria dos Grafos
Lista de exercícios 01

Questão 1. Seja G um grafo, T uma árvore geradora de G e $v \in V(G)$.

a) É verdade que se v tem grau maior que 1 em T, então v é vértice de corte em G? Justifique.

b) É verdade que se v é vértice de corte em G então v tem grau maior que 1 em T? Justifique.

c) Um vértice de corte em um grafo pode ser folha de uma árvore geradora deste grafo? Justifique.

Questão 2. Prove que todo grafo conexo com n vértices e $n-1$ arestas é árvore.

Questão 3. Prove que um grafo T é árvore geradora de um grafo G se e somente se T é subgrafo conexo minimal de G.

Questão 4. Prove que toda árvore T tem (pelo menos) $\Delta(T)$ folhas.

Questão 5. Prove que, se $G = (V, E)$ é um grafo direcionado, então

$$\sum_{v \in V} \delta^+(v) = |A(G)| = \sum_{v \in V} \delta^-(v).$$

Questão 6. Considere a ideia de evitar o desperdício de espaço na representação da matriz de adjacência M_G de um grafo $G = (V, E)$ com $V = \{1, \ldots, n\}$ usando um vetor m contendo somente os elementos acima da diagonal principal de M_G, de tal maneira que

$$M_G(u, v) = \begin{cases} m[f(u, v)], & se\ u < v, \\ 0, & se\ u = v, \\ m[f(v, u)], & se\ u > v, \end{cases}$$

onde $f = \{(u, v) \in V \times V | u < v\} \to \{0, \ldots, N(n) - 1\}$ é a função que faz a correspondência entre os elementos de M_G e m, e $N(n)$ é o tamanho do vetor m.

a) Dê uma expressão para $N(n)$.

b) Dê uma expressão para $f(u, v)$.

c) Implemente um programa que utiliza a forma de armazenamento descrito acima para ler os dados de um grafo e imprimir para cada vértice os elementos adjacentes a ele.

d) Escreva a função

```
unsigned int vizinho(unsigned int *m, unsigned int u, unsigned int v);
```

que devolve o valor de $M_G(u, v)$ é representado pelo vetor m tal como descrito acima.

Questão 7. Seja G um grafo com n vértices e m arestas. Dê uma expressão para

a) O número de subgrafos de G induzidos por vértices.

b) O número de subgrafos de G induzidos por arestas.

c) O número de subgrafos geradores de G.

d) O número de subgrafos de G (não precisa ser em termos de n e m).

Questão 8. Sobre o problema da árvore geradora mínima, implemente um programa para testar os algoritmos de Prim e de Kruskal. Seu programa deve ser em linguagem C.

Questão 9. Considere $C(T)$ o conjunto de componentes conexas de um grafo T. Se T é uma árvore, então

$$|C(T - v)| = \delta_T(v),$$

para todo $v \in T$.

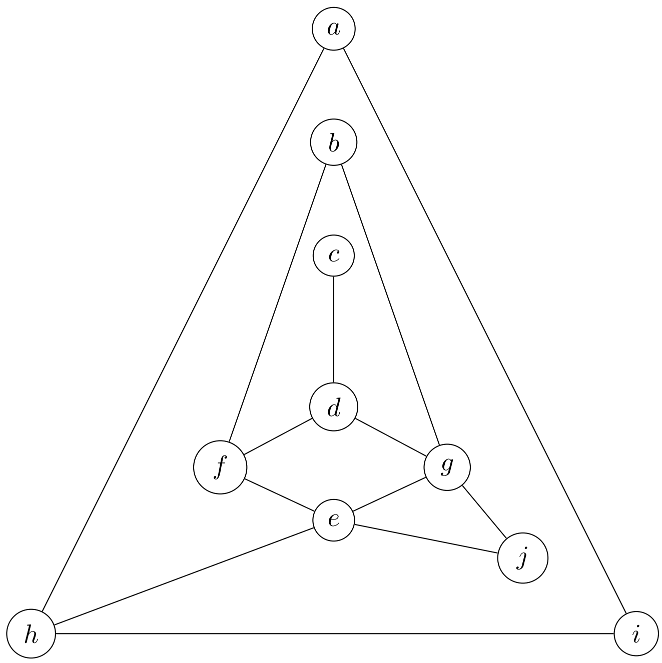
Questão 10. Prove que se T é a arborescência resultante de uma busca em largura a partir do vértice r em um grafo conexo G, então rTv é um caminho mínimo em G para todo $v \in V(G)$.

Questão 11. Caracterize

a) as árvores em largura de um grafo completo, e

b) as árvores em profundidade de um grafo completo.

Questão 12. Execute uma busca em profundidade no grafo abaixo a partir do vértice a.



a) Mostre os valores dos índices de pré-ordem, pós-ordem de cada um dos vértices.

b) Indique os vértices e arestas de corte do grafo e explique como decidir que o são com base nestes valores.

Questão 13. Seja T uma arborescência geradora de um grafo G produzida por uma busca em profundidade. Prove que

a) A raiz de T é vértice de corte de G se e somente se tem mais de um filho.

b) Um vértice v que não é raiz de T é vértice de corte de G se e somente se tem um filho w tal que nenhum descendente de w é vizinho de um ancestral próprio de v.

c) Uma aresta (u, v) é aresta de corte em G se e somente se $(u, v) \in A(T)$ e nenhuma aresta em G liga um descendente de v a um ancestral de u.

Questão 14. Implemente um algoritmo que calcula o número de componentes conexas de um grafo $G = (V, E)$ fornecido.

Questão 15. Implemente um algoritmo para determinar se um dado grafo $G = (V, E)$ é bipartido ou não.

Questão 16. Implemente um algoritmo para determinar se um dado grafo $G = (V, E)$ é orientável ou não.

Questão 17. Denote por $\chi(G)$ o número cromático de um grafo G. Prove que todo grafo G com m arestas satisfaz:

$$\chi(G) \leq \frac{1}{2} + \sqrt{2m + \frac{1}{4}}.$$