

Primeira Prova de Teoria dos Grafos. Turma A.
Universidade Federal de São Carlos
Prof. José de Oliveira Guimarães.

Observações:

- todos os grafos são conectados e não dirigidos a menos que o exercício diga o contrário;
- prove todas as suas afirmações ou cite um teorema que você conhece para prová-las.

1. (2.5) Prove: se o grau de cada vértice de um grafo G é pelo menos dois, então G contém pelo menos um ciclo.

2. (2.5) Calcule a complexidade dos seguintes algoritmos:

a) Dado um grafo $G = (V, E)$, o algoritmo $adj(v, w)$ retorna true se o vértice v é adjacente a w ; isto é, há uma aresta entre v e w . Aqui os vértices são números e o algoritmo utiliza uma lista de adjacências para representar o grafo. A lista é um vetor indexado pelo número do vértice onde cada elemento aponta para uma lista ...

b) Dado um grafo $G = (V, E)$, o algoritmo $normal(G)$ retorna true se o grafo G não possui vértices com arestas para eles mesmos. O grafo é representado por uma matriz de adjacências. Lembre-se de que o tamanho da entrada não é $|V| + |E|$ mas ... (resposta abstraída ...).

3. (2.5) Faça um algoritmo para o seguinte problema: dado um grafo acíclico dirigido, numere os vértices de 1 até n tal que, se v possui número k , então todos os vértices que podem ser atingidos por um caminho dirigido possuem números $> k$. Este é o algoritmo de ordenação topológica. Comece pelo caso base: como numerar um grafo com um (1) vértice? Qual é a Hipótese de Indução? Como remover um vértice para chegar à HI? Você pode utilizar o seguinte lema na sua resposta: em um grafo acíclico dirigido, há pelo menos um vértice com grau de entrada 0. Naturalmente, este vértice desempenha um papel fundamental no algoritmo de ordenação topológica.

4. (2.5) Uma árvore é um grafo conectado sem ciclos. Prove que há um único caminho entre dois vértices quaisquer de uma árvore.

5. (2.0) Explique como funciona o algoritmo de busca em largura sem escrever o seu código. Faça um exemplo e diga qual estrutura de dados o algoritmo utiliza. Não é necessário justificar.