PUC-Rio
Departamento de Informática
Profs. Marcus Vinicius S. Poggi de Aragão
Período: 2004.1
Horário: 2as-feiras e 4as-feiras de 19-21 / 21-23 horas
16 de junho de 2004

ESTRUTURAS DISCRETAS (INF 1308)

3^a Lista de Exercícios

Data da Entrega: 30 de junho de 2004

Procure ser conciso e preciso nas suas argumentações.

1. Seja $G = (V, E)$ um grafo orientado. Representa-se por d_{ij} a distância do vértice i ao vértice j para todo arco (i, j) em E, sendo que d_{ij} pode assumir valor negativo, mas o grafo não possui circuito negativo.

   Seja agora uma árvore geradora de G, $T = (V, F)$ com raiz em um vértice r.

   Proponha um algoritmo para determinar se os caminhos de r aos demais vértices sobre T são (todos) os caminhos mais curtos no grafo G ou não.

   Ou seja, o algoritmo deve responder SIM, se todos forem caminhos mais curtos, ou NÃO caso um dos caminhos não seja um caminho mais curto. O seu algoritmo deve fazer no máximo $2.|E|$ comparações.

   Dica: Verifique as condições para que um caminho seja mais curto. Quais são ?

2. O fecho transitivo de um grafo orientado $G = (V, E)$ é uma matriz M $n \times n$ ($n = |V|$) onde a posição $M(i, j)$ desta matriz indica se existe (1) ou não (0) um caminho (orientado) do vértice i para o vértice j. Proponha um algoritmo para determinar o fecho transitivo de $G = (V, E)$, orientado.

   Dica: Utilize um algoritmo que permita determinar os vértices em que é possível chegar a partir de um vértice inicial dado.

3. Dado um grafo não-orientado $G = (V, E)$ onde todos os vértices possuem grau par, proponha um algoritmo para orientar as arestas do grafo de modo que o grau de entrada (número de arcos que chegam no vértice) seja igual ao grau de saída (número de arcos que saem do vértice) em todos os vértices do grafo.

   Dica: Lembre que em um ciclo orientado todos os vértices tem o número de arcos entrando igual ao número de arcos saindo.

4. Dado um grafo orientado $G = (V, E)$ com distâncias associada aos arcos d_{ij} para $(i, j) \in E$, proponha um algoritmo para determinar o caminho mais curto do vértice s ao vértice t que utilize exatamente k arcos. Observe que este caminho pode repetir vértices ou mesmo arcos, ou seja não é necessariamente um caminho simples.

   Dica: Em uma busca em largura determina-se a cada etapa os vértices que estam a um determinado número de arestas do vértice de partida.

5. Considere um grafo não orientado $G = (V, E)$ onde w_{ij} corresponde ao peso da aresta (i, j) para todas as arestas em E. Considere também uma floresta em G, $f = (V, F)$ onde $F \subset E$.

   Proponha um algoritmo para obter a árvore geradora de G que contém todas as arestas em F e possui peso mínimo.

   Dica: Adapte um dos algoritmo que você conhece para obter árvore geradora mínima.

6. Considere o problema de encontrar o caminho-mais-curto de s a todos os demais em V.

   (a) Prove o teorema para K assumindo valores $1, 2, \ldots, |V|$ para qualquer grafo $G = (V, A)$ com distâncias associadas aos arcos d_{ij} para $(i, j) \in A$.

   Teorema 1 *(K): Sabe-se determinar o K-ésimo vértice mais próximo a s e sua respectiva distância mínima de s.*

   (b) Explique a sua prova utilizando como exemplo o grafo $G = (V, A)$ abaixo fazendo $s = 1$. $V = \{1, 2, 3, 4, 5, 6\}$, $A = \{(1, 2), (1, 3), (3, 5), (3, 6), (3, 4), (2, 4), (4, 5), (4, 6)\}$, com distâncias $d_{1,2} = 1, d_{1,3} = 2,\ d_{3,5} = 4, d_{3,6} = 2, d_{3,4} = 2,\ d_{2,4} = 2, d_{4,5} = 2, d_{4,6} = 3$.