UFMG/ICEx/DCC — Projeto e Análise de Algoritmos

# Lista de Exercícios 3

Pós-Graduação em Ciência da Computação — 1º Semestre de 2009

As questões, a seguir, tratam do projeto de algoritmos. CLSR é a referência do exercício no livro-texto. As questões estão em inglês por terem sido copiadas da versão original.

**Questão 1 [CLRS, Ex 22.1-8, pg 531]**

Suppose that instead of a linked list, each array entry $Adj[u]$ is a hash table containing the vertices $v$ for which $(u, v) \in E$. If all edge lookups are equally likely, what is the expected time to determine whether an edge is in the graph? What disadvantages does this scheme have? Suggest an alternate data structure for each edge list that solves these problems. Does your alternative have disadvantages compared to the hash table?

**Questão 2 [CLRS, Ex 22.2-7, pg 539]**

The diameter of a tree $T = (V, E)$ is given by

$$\max_{u,v \in V} \delta(u, v),$$

that is, the diameter is the largest of all shortest-path distances in the tree. Give an efficient algorithm to compute the diameter of a tree, and analyze the running time of your algorithm.

**Questão 3 [CLRS, Ex 22.3-12, pg 549]**

A directed graph $G = (V, E)$ is *singly connected* if $u \rightsquigarrow v$ implies that there is at most one simple path from $u$ to $v$ for all vertices $u, v \in V$. Give an efficient algorithm to determine whether or not a directed graph is singly connected.

**Questão 4 [CLRS, Ex 22.5-3, pg 557]**

Professor Deaver claims that the algorithm for strongly connected components can be simplified by using the original (instead of the transpose) graph in the second depth-first search and scanning the vertices in order of increasing finishing times. Is the professor correct?