max planck institut informatik

Selected Topics in Algorithms
K. Mehlhorn
Exercise 6

Summer 2009
We will discuss this exercise sheet on June 26th and June 29th.

Certifying Gaussian Elimination

Let A be an $m \times n$ matrix with rational entries and b be a $m \times 1$ vector with rational entries. Gaussian elimination decides whether the linear system $Ax = b$, where x is a $n \times 1$ vector of unknowns, is solvable. If the system is solvable, it computes a vector $z \in \mathbb{Q}^n$ such that $Az = b$. If the system is unsolvable, Gaussian elimination declares this fact. Extend Gaussian elimination such that, in the latter case, it returns a $m \times 1$ vector $c \in Q^m$ with $c^T A = 0$ and $c^T b \neq 0$.

Coloring Interval Graphs

Let $\mathscr{I}$ be a collection of intervals with real endpoints. A coloring of $\mathscr{I}$ is a function $c : \mathscr{I} \to \mathbb{N}$ such that $f(I_1) \neq f(I_2)$ whenever I_1 and I_2 intersect. A coloring is optimal if it uses a minimal number of colors.

1. A subset $\mathscr{C}$ of $\mathscr{I}$ is called a *clique* if any two intervals in $\mathscr{C}$ intersect. Prove: if $\mathscr{C}$ is a clique in $\mathscr{I}$, then any coloring of $\mathscr{I}$ uses at least $|\mathscr{C}|$ colors.
2. Show: if $\mathscr{C}$ is a clique, then there is a real x with $x \in I$ for all $I \in \mathscr{C}$.
3. Design an algorithm that computes an optimal coloring. You may assume for simplicity that the endpoints of the intervals are pairwise distinct. Prove the correctness of your algorithm. What is its running time?
4. Make your algorithm certifying. For example, if your coloring uses the colors 1 to k, it might output a clique of size k.
5. Remove the restriction that endpoints are distinct.

Strongly Connected Components

A directed graph $G = (V,E)$ is *strongly connected* if for any two nodes v and w of G, there is a path from v to w and a path from w to v.

1. Find a nice witness for a graph being strongly connected.
2. Find a nice witness for a graph being not strongly connected.
3. Design linear time algorithms for finding such witnesses.