Winter Term 2001/02 | Prof. Dr. Susanne Albers
October 22, 2001 | Institute of Computer Science

Algorithms Theory

Problem Set 1

To be submitted on November 5, 2001 (in the lecture)

Problem 1: (10 points)
In this course we will often be interested in the running times of algorithms. Assume that an algorithm A uses $f(n)$ milliseconds to solve a problem of size n. In the table below determine, for given $f(n)$ and time t the maximum problem size that can be solved in time t.

	1 second	1 minute	1 hour	1 day
$\log n$				
$\sqrt{n}$				
n				
$n \log n$				
n^2				
n^3				
2^n				

Problem 2: (10 points)
Given a directed acyclic graph $G = (V, E)$, a *topological sort* of G is a function $label : V \longrightarrow \{1, \ldots, n\}$, such that, for any edge $(v, w) \in E$, $label(v) < label(w)$. Prove that the following algorithm computes a correct topological sort. What is the running time of the algorithm?

Algorithm TOPSORT
1. $i := 0$; $j := |V| + 1$;
   for all $v \in V$ **do** $num(v) := 0$; $label(v) := 0$; **endfor**;
2. **for all** $v \in V$ **do**
      if $num(v) = 0$ **then** TOPSORT(v) **endif**;
   endfor;

Procedure TOPSORT(v)
1. $i := i + 1$; $num(v) := i$;
2. **for all** $(v, w) \in E$ **do**
3.   **if** $num(w) = 0$ **then**
4.   TOPSORT(w);
5.   **else if** $label(w) = 0$ **then** STOP "no topological sort possible" **endif**;
6. **endfor**;
7. $j := j - 1$; $label(w) := j$;

Problem 3: (8 + 2 points)

Consider the refined shortest paths algorithm presented in the lecture.

a) Modify the algorithm such that, for acyclic networks, it always chooses a node $u \in U$ with $\text{DIST}[u] = dist(s, u)$ in line 5 of the algorithm. Show that your solution is correct.

b) Modify the algorithm such that the actual shortest paths are computed, too.

Problem 4: (9 + 1 points)

Consider again the refined shortest paths algorithm and suppose that an arbitrary (i.e. not necessarily acyclic) network is given. Assume that the set U is implemented as a standard queue.

a) Prove the following lemma:

> If the network does not contain negative cycles, each node is removed at most n times from the queue U.

b) How could you use this lemma to detect negative cycles?

Information on the Course

The latest information on the course, including the problem sets, can be found at

`http://www.informatik.uni-freiburg.de/~salbers/lehre/algo.html`

The solutions of the problem sets may submitted in teams of up to two students.

Credit points: To earn credit points students must

- complete successfully 50% of the homework assignments
- present their solutions twice in the problem sessions and
- pass a final exam.