Comprehensive Exam: Algorithms and Concrete Mathematics

Autumn 2002

This is a one hour closed-book exam and the point total for all questions is 100.

1. *(10 points)* Let $f(n) = 2^{\sqrt{\log n}}$ and $g(n) = n^{\epsilon}$, for some constant ϵ. Which one of the following claims is true:

   (a) $f(n) = O(g(n))$

   (b) $f(n) = o(g(n))$

   (c) $f(n) = \Omega(g(n))$

   (d) $f(n) = \omega(g(n))$

   Prove your answer.

2. *(15 points)* Prove a tight asymptotic bound on the behavior of $T(n)$.
   $T(n) = T(\lceil 0.4n \rceil) + T(\lceil 0.5n \rceil) + n$, where for $k \leq 10$ we are given that $T(k) = \Theta(1)$. Do not disregard the "ceiling" operations, i.e. provide a derivation that takes them into account or explicitly shows that they do not change the result.

3. *(20 points)* Prove that the following algorithm generates a random permutation of the numbers $1, 2, \ldots, n$:

```
Set A[i] <- i;
For i := 1,2,...,n :
   j <- Random uniformly distributed integer in range [i,n];
   Swap(A[i],A[j]);
endfor
```

   In particular, prove for all i, j that $\Pr(A[i] = j) = 1/n$.

4. *(20 points)* You are given a graph $G(V, E)$ with n nodes and two length functions: $l_1 : E \to \{1, 2, \ldots, n\}$ and $l_2 : E \to \{1, 2, \ldots, n\}$.

   Given two integers a and b, and given two nodes s and t, give an algorithm to check if there is a path from s to t whose length according to the function l_1 is at most a, and whose length according to the second function l_2 is at most b. Prove the running time of your algorithm and show that it is polynomial in n. [Hint: make use of the fact that lengths are integers in a polynomial range.]

5. (*15 points*) A dominating set in a graph $G(V, E)$ is a set of vertices $S \subseteq V$ such that every vertex in $V - S$ is adjacent to at least one vertex in S.

Consider the following greedy algorithm for dominating set: Find the vertex with largest degree. Include it in the dominating set, and delete it and all its neighboring vertices from the graph. Again find the vertex with largest degree in the remaining graph, include it in the dominating set, delete it and its neighbors from the graph, and so on. Repeat till the graph is empty.

Show that there exist graphs with n vertices that have $O(\sqrt{n})$-size optimum dominating set and where the above algorithm produces a (much worse) $O(n)$-size dominating set. Partial credit for any other graph families that show that the above algorithm is not good.

6. (*20 points*) Suppose you are allowed only equality comparisons among objects. In other words, there is no ordering defined over the objects; you can only check if two objects are equal or not. You are given n objects in an array. Show an $O(n)$-time deterministic algorithm to find out if there is an object appearing strictly more than $n/2$ times in the array. For simplicity, you may assume n is a power of 2. 10 points for a randomized algorithm.