Aalto University
Department of Information and Computer Science
Pekka Orponen

T-79.4202 Principles of Algorithmic Techniques (5 cr)
Exam Thu 20 Feb 2014, 1–4 p.m.

Write down on each answer sheet:
- Your name, degree programme, and student number
- The text: "T-79.4202 Principles of Algorithmic Techniques 20.2.2014"
- The total number of answer sheets you are submitting for grading

Note: You can write down your answers in either Finnish, Swedish, or English.

1. Arrange the following functions according to their increasing order of growth:

$$\sqrt{n}, \quad n\log n, \quad n^{1/3}+\log n, \quad \log n, \quad e^n,$$
$$n!, \quad 42, \quad \log\log n, \quad (1/e)^n, \quad n/\log n.$$

(Notation $\log n$ denotes here logarithm in base 2.)

2. You need to make a choice between two alternative algorithms for a given problem. Algorithm A partitions a problem instance of size n into two subinstances of size $n/2$, each of which is solved recursively, and the partitioning and combining of the subsolutions takes time $O(n)$. Algorithm B on the other hand uses time $O(n^2)$ to partition the given problem instance into three subinstances of size $n/3$, but only *one* of these is then solved recursively. Which one of the algorithms is asymptotically faster? Justify your answer.

3. Design an algorithm $B(n,k)$ that on inputs n and k prints out those binary sequences of length n that contain exactly k ones. The running time $T(n,k)$ of your algorithm must be proportional to the number of binary sequences it outputs, i.e. $T(n,k) = O(\binom{n}{k}) = O(n^k)$ (for a fixed value of $k \geq 1$). Present the time complexity analysis of your algorithm.

4. In the NP-complete *Independent Set* problem one is given as input an undirected graph $G = (V,E)$, and the task is to find as large as possible subset of vertices $V' \subseteq V$ with no interconnecting edges, i.e. satisfying the condition $u,v \in V' \implies \{u,v\} \notin E$. Design some reasonable polynomial-time approximation algorithm for this problem, and present an example of a graph G where your algorithm does *not* conclude with an optimal solution, i.e. a maximally large independent set of vertices.

Grading: Each problem 12p, total 48p.