Aalto University School of Science and Technology 18 May 2010
Dept. of Communications and Networking, Dept. of Information and Computer Science

Exam: S-72.2420/T-79.5203 Graph Theory

Include the following on every answer sheet: name of the course, THE COURSE CODE FOR WHICH YOU WANT A MARK, the date, your name, your student ID, and YOUR DEGREE PROGRAMME.

1. Proofs.

   (a) (3p.) Prove or disprove: Let G be an Eulerian graph and H be an Eulerian subgraph of G. Then the graph obtained by deleting the edges of H from G is also Eulerian.

   (b) (3p.) The Ramsey number $R(s,t)$ is the least integer n such that all graphs on n vertices contain an s-vertex clique or a t-vertex independent set. Show that $R(s,t) \leq R(s-1,t) + R(s,t-1)$.

2. Tournament graphs are oriented complete graphs, that is, for each pair of distinct vertices v, w, either vw or wv is an edge (but not both).

   (a) (3p.) Show that all tournaments contain a directed Hamiltonian path.

   (b) (3p.) Show that all tournaments on $n \geq 2$ vertices have a dominating set with $\lceil \log_2(n+2) \rceil - 1$ vertices. ($S \subseteq V$ is a dominating set, if for all $v \in V \setminus S$ there exists $u \in S$ such that uv is an edge.)

3. Vertex r is a *root* of digraph G, if there exists a path from r to v for each $v \in V(G)$. (Note that a digraph may have several roots, or none at all.)

   (a) (3 p.) Design an algorithm that finds a root in a digraph G, represented as an adjacency list, in time $O(n(G) + e(G))$, or determines that none exist. (*Hint:* A constant number of DFS passes over G suffices.)

   (b) (3 p.) Design an efficient algorithm for determining *all* roots of a given digraph G, and analyze its worst-case running time.

4. (6 p.) *Currency arbitrage* means "taking advantage of divergences in exchange rates in different money markets by buying a currency in one market and selling it in another market" (`http://www.thefreedictionary.com`). E.g. if one euro buys 1.25 US dollars, one US dollar 7.80 Swedish crowns, and one Swedish crown 0.12 euros, then by exchanging first euros into dollars, then dollars into crowns, and finally crowns back into euros, one profits at a rate of $1.25 \times 7.80 \times 0.12 = 1.17$. (Presumably such an advantageous set of exchange rates needs to be collected across several different markets, but we will ignore this point.) Design an efficient algorithm that determines, given a system of n currencies with exchange rate of currency i against currency j given by r_{ij}, whether profitable arbitrage is possible. Provide an upper bound on the worst-case running time of your algorithm. (It must of course be polynomial in n.)