Aalto University School of Electrical Engineering
Department of Communications and Networking

EXAM
4.1.2011

S-72.2420 Graph Theory

Include the following on every answer sheet: name of the course, the course code, the date, your name, your student ID, and your DEGREE PROGRAMME.

1. (6p.) Basic terminology. Draw (with motivations)
   (a) the 3-dimensional cube, Q_3;
   (b) a path with six vertices, P_6;
   (c) a connected graph of order 5 that has exactly one induced cycle of length 4 (C_4);
   (d) a graph of order 5 that has a perfect matching if any of the vertices (and its incident edges) is removed;
   (e) a graph G that has independence number $\alpha(G) = 3$ and clique number $\omega(G) = 3$;
   (f) a graph G with chromatic number $\chi(G) = 2$.

2. (6p.) Graph-theoretical proofs.
   (a) (2p.) Let G be a connected graph with n vertices, and let k be a positive integer less than n. Is it always true that there is a k-element subset X of $V(G)$ such that $G - X$ is connected?
   (b) (2p.) Characterize the graphs with no induced subgraph isomorphic to $K_{1,2}$.
   (c) (2p.) Either draw two nonisomorphic 10-vertex, 4-regular, bipartite graphs, or prove that there is only one such graph.

3. (6p.) Matchings.
   (a) (3p.) A *permutation matrix* is a matrix with entries in $\{0, 1\}$ such that every row and every column has a unique 1-entry. Let A be a matrix with nonnegative integer entries a_{rc} such that the entries in every row r and every column c sum to k, $k \geq 1$. Prove that there exist permutation matrices $P_1, P_2, \ldots, P_k$ such that $A = P_1 + P_2 + \ldots + P_k$.
   Hint: Let r and c be joined by a_{rc} edges.
   (b) (3p.) Prove or disprove: There is, up to isomorphism, only a finite number of *connected* 3-regular simple graphs with no 1-factor.

4. (6p.) Spanning trees with forced leaves. Design a polynomial-time algorithm for the following problem. The input consists of (1) an undirected graph G; (2) a set of vertices $U \subseteq V(G)$; and (3) a weight $w(e) \geq 0$ for each edge $e \in E(G)$. The task is to either (i) output a spanning tree T of G such that each $u \in U$ is a leaf of T and the total weight $w(T) = \sum_{e \in E(T)} w(e)$ is the minimum possible over all such spanning trees; or, (ii) assert that the graph G has no spanning tree meeting the requirements. Carefully justify the correctness of your algorithm.