Technische Universität München
Department of Computer Science
Chair for Efficient Algorithms
Prof. Dr. Sami Khuri

WS 1999/2000
Problem Set 6
June 6, 2000

Efficient Algorithms and Data Structures II

Problem 1 (10 credits)

Show that a graph is bipartite if and only if it has no odd length cycles.

Problem 2 (10 credits)

Consider the following greedy algorithm for finding a maximal matching: Start with an arbitrary edge as the initial matching. Find another edge that does not have a vertex in common with the current matching. If one does exist, add it to the current matching. Repeat until no more edges can be added.

- **a.** What is the running time of this algorithm on a graph with n vertices and m edges?
- **b.** Give an example of a graph where the algorithm does not find a maximum matching.
- **c.** Show that the matching found by the algorithm always has at least half as many edges as a maximum matching.

Problem 3 (10 credits)

Recall that a k-regular graph is a graph in which each vertex has degree k.
Show that every k-regular bipartite graph $G = (V, E)$ with vertex partition $V = L \cup R$ and $k > 0$ has a complete matching.
(*Hint: Use Hall's theorem for bipartite graphs.*)

Problem 4 (10 credits)

Let $G = (V, E)$ be a bipartite graph with vertex partition $V = L \cup R$, and let G' be its corresponding flow network. Give a good upper bound on the length of any augmenting path found in G' during the execution of Ford-Fulkerson.