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

Summer semester 2000
Problem Set 8
July 18, 2000

Efficient Algorithms and Data Structures II

Problem 1 (15 credits)

Let G be an undirected, bipartite graph with real edge weights.

a. Devise an efficient algorithm that determines a maximum cardinality maximum weight matching.

b. Devise an efficient algorithm that determines a maximum weight matching.

Problem 2 (15 credits)

a. For a graph G let $\mathcal{C}_G$ denote the set of connected components, and let $q(G)$ denote the number of components with odd cardinality.

> **Theorem** (Tutte, 1947): A graph $G = (V, E)$ has a perfect matching if and only if $q(G - S) \leq |S|$ for all $S \subseteq V$.

Show that the condition of the theorem is necessary for the existence of a perfect matching.

b. A graph $G = (V, E)$ is called *factor critical* if $V \neq \emptyset$ and $G - \{v\}$ has a perfect matching for any $v \in V$. A vertex set $S \subseteq V$ is called *pairable with* $G - S$, if the graph H_S, which is obtained from G by contraction of all components in $\mathcal{C}_{G-S}$ and deletion of all edges within S, contains a matching of cardinality $|S|$.

Here, under contraction of a subgraph $S \subseteq G$, we understand the consecutive elimination of all edges within S and amalgamation of the endpoints of any eliminated edge, until just one vertex remains (which then is adjacent to all nodes $V - S$ that have been adjacent to any node of S before).

> **Theorem**: Each graph $G = (V, E)$ contains a vertex set S with the following properties.
>
> 1. S is pairable with $G - S$.
> 2. Each component of $G - S$ is factor critical.
>
> For each such S, G contains a perfect matching if and only if $|S| = |\mathcal{C}_{G-S}|$.

Show that Tutte's theorem follows from this theorem.

Problem 3 (10 credits)

Design a linear time algorithm to determine if a string T is a cyclic rotation of another string T'. For example, *particle* and *ticlepar* are cyclic rotations of each other. (Hint: Use the KMP algorithm.)