Graphalgorithmen — Sommersemester 2021

Prof. Dr. Johannes Köbler — 24. Juni 2021

Übungsblatt 10

Besprechung der mündlichen Aufgaben ab 1. 7. 2021
Abgabe der schriftlichen Lösungen bis 8. 7. 2021, 13:00 Uhr

Aufgabe 42 ***mündlich***

Wie groß ist die Anzahl $\alpha(n)$ aller maximalen Matchings und die Anzahl $\beta(n)$ aller Matchings im vollständigen Graphen K_n mit n Knoten?

Aufgabe 43 ***mündlich***

(a) Zeigen Sie, dass ein Baum höchstens ein perfektes Matching hat.

(b) Entwerfen Sie einen Algorithmus, der in Linearzeit ein maximales Matching für einen gegebenen Baum berechnet.

Aufgabe 44 ***mündlich***

Gegeben sind k Personen $A_1, \ldots, A_k$ und ℓ Maschinen $M_1, \ldots, M_\ell$ sowie eine Menge $E \subseteq V_1 \times V_2$ von Jobs, wobei $V_1 = \{1, \ldots, k\}$ und $V_2 = \{1, \ldots, \ell\}$ ist.

(a) Zeigen Sie, dass der bipartite Graph $G = (V_1, V_2, E)$ ein Matching M hat, in dem kein Knoten u vom Grad $\deg(u) = \Delta(G)$ frei bleibt.

Hinweis: Konstruieren Sie M aus 2 Matchings M_i ($i = 1, 2$), die keinen Knoten $u \in V_i$ vom Grad $\deg(u) = \Delta(G)$ frei lassen. Bei der Konstruktion von M_i können Sie Aufgabe 39 verwenden.

(b) Bestimmen Sie die Zeit $t(E)$, die zur Erledigung aller Jobs in E benötigt wird, falls jeder Job $(i, j) \in E$ in einer Zeiteinheit und zwar nur von Person A_i an Maschine M_j erledigt werden kann und jede Person und jede Maschine in jeder Zeiteinheit maximal einen Job übernehmen kann. *Hinweis:* Teilaufgabe (a).

(c) Finden Sie einen Algorithmus, der in Zeit $\mathcal{O}(m\Delta(G)\sqrt{\min(k, l)})$ einen Zeitplan zur Erledigung aller Jobs in E in $t(E)$ Zeiteinheiten erstellt, wobei $m = |E|$ ist.

(d) Was ändert sich, wenn E eine Multimenge ist?

Aufgabe 45 ***mündlich***

Zeigen Sie, dass der Algorithmus von Dinitz das Matchingproblem für bipartite Graphen G unter Verwendung von `blockfluss1` in Zeit $O(m\sqrt{\mu})$ löst (s. Hinweis zu Aufgabe 35).

Aufgabe 46 ***10 Punkte***

(a) Finden Sie einen Algorithmus A, der für einen bipartiten Graphen $G = (U, W, E, k)$ mit Kostenfunktion $k : E \to \mathbb{Z}$ in Zeit $O(mn \log n)$ für $i = 1, \ldots, \mu(G)$ ein kostenminimales Matching M_i der Größe i berechnet.

Zudem soll A für jedes Matching M_i eine Preisfunktion p_i berechnen, so dass die reduzierten Kosten $k^{p_i}(u, w)$ für alle Kanten $\{u, w\} \in E$ nichtnegativ sind und für alle $\{u, w\} \in M_i$ den Wert 0 haben.

(b) Beschreiben Sie die Arbeitsweise Ihres Algorithmus bei Eingabe des folgenden Graphen $G = (U, W, E, k)$: