Fakultät für Informatik
Lehrstuhl 11
Prof. Dr. Petra Mutzel
Dipl.-Inf. Till Schäfer

Übungsblatt 11

Besprechungszeit:
01.-05.07.2013
KW27

Übungen zur Vorlesung

Effiziente Algorithmen

Sommersemester 2013

Bitte beachten Sie die Hinweise zu den Übungen und der Übungsgruppenverteilung auf der Homepage der Übung.

Die (freiwilligen) schriftlichen Lösungen können Sie einfach in Ihrer Übungsgruppe abgeben (gerne auch als Gruppenabgaben).

Aufgabe 11.1 – Wiederholung

- Wie berechnet sich die erwartete Rechenzeit bei randomisierten Algorithmen?
- Wie ist das SAT-Problem definiert und was ist das Max-k-SAT-Problem? Welche Algorithmen haben Sie für das Max-k-SAT-Problem in der Vorlesung kennengelernt? Wie sind die Laufzeiten und die Gütegarantien?

Aufgabe 11.2 – 2-SAT

Beschreiben Sie einen deterministischen Algorithmus, der die Erfüllbarkeit einer 2-SAT-Instanz mit m Klauseln und n Variablen in Laufzeit $\mathcal{O}(n+m)$ überprüft.

Hinweis: Bekanntermaßen ist $a \vee b$ äquivalent zu $\bar{a} \Rightarrow b$. Daher ist es sinnvoll, zu einer gegebenen 2-SAT-Instanz $\mathcal{I}_{2-SAT} = (U, C)$ mit einer Variablenmenge U und einer Klauselmenge C den wie folgt konstruierten Graphen $G_{\mathcal{I}} = (V, E)$ zu betrachten, der die in den Klauseln enthaltenen Implikationsbeziehungen erfasst: Die Knotenmenge ist $V = \{x_1, \bar{x}_1, x_2, \bar{x}_2, \ldots, x_n, \bar{x}_n\}$ und die Kanten ergeben sich daraus, dass man für jede Klausel $(a \vee b) \in C$ die zwei Kanten $(\bar{a}, b), (\bar{b}, a) \in V \times V$ aufnimmt.

Zeigen Sie, dass eine Instanz $\mathcal{I}_{2-SAT}$ genau dann nicht erfüllbar ist, wenn es eine Variable $x \in U$ gibt, so dass x und $\bar{x}$ in der gleichen starken Zusammenhangskomponente von $G_{\mathcal{I}}$ liegen und nutzen Sie diese Beobachtung in Ihrem Algorithmus aus.

Aufgabe 11.3 – PARTITION

Wir betrachten das NP-vollständige Entscheidungsproblem PARTITION: Für n natürliche Zahlen $b_1, \ldots, b_n$ ist zu entscheiden ob es eine Teilmenge $M \subset \{1, \ldots, n\}$ gibt, so dass $\sum_{i \in M} b_i = \sum_{i \notin M} b_i$ gilt.

Geben Sie einen pseudopolynomiellen Algorithmus für PARTITION an, d.h. einen Algorithmus, dessen Rechenzeit polynomiell in den Zahlen n und $\sum_{i=1}^{n} b_i$ ist.

Aufgabe 11.4 – Berechnung eines Eulerkreises

Sei $G = (V, E)$ ein Multigraph mit n Knoten und m Kanten, der durch seine Adjazenzlistendarstellung gegeben ist. Finden Sie einen Algorithmus, welcher entscheidet, ob G einen Eulerkreis besitzt und, falls ja, einen solchen berechnet. Die Laufzeit Ihres Algorithmus soll durch $O(n + m)$ beschränkt sein.