Übungsblatt 12

Algorithmen I – Sommersemester 2022

Abgabe im ILIAS bis 20.07.2022, 14:00 Uhr

Bitte beschrifte Deine Abgabe gut sichtbar mit Deinem Namen und Deiner Matrikelnummer. Achte insbesondere bei handschriftlichen Abgaben auf Lesbarkeit und genügend Platz für Korrektur-Anmerkungen. Die Abgabe erfolgt über das Übungsmodul in der Gruppe Deines Tutoriums im ILIAS. Gib Deine Ausarbeitungen in *einer* PDF-Datei ab. Achte darauf, effiziente Algorithmen zu formulieren, also solche mit möglichst geringer asymptotischer Laufzeit!
Wenn du die Korrektheit eines Algorithmus begründen oder dessen Laufzeit analysieren sollst, tue dies getrennt von der Beschreibung des Algorithmus.
Wenn nicht anders spezifiziert oder aus dem Kontext ersichtlich, bezeichnen wir mit Graph einen einfachen ungerichteten Graphen.

Aufgabe 1 - Spannend, Bäume! (10 Punkte)

Sei $G = (V, E)$ ein zusammenhängender, gewichteter Graph, bei dem keine zwei Kanten das gleiche Gewicht haben. Wir wollen uns in dieser Aufgabe mit Spannbäumen von G befassen.

1. Sei $C \subset V$ ein Kreis in G. Beweise, dass es keinen minimalen Spannbaum von G gibt, der die schwerste Kante von C enthält. (3 Punkte)

2. Beschreibe einen Algorithmus, der für G einen Spannbaum mit *maximalem* Gewicht bestimmt. Begründe die Korrektheit des Algorithmus und zeige, dass er die asymptotische Laufzeit von Prim / Kruskal nicht überschreitet. (4 Punkte)

3. Wir nennen eine Kantenmenge $X \subset E$ ein *Feedback Edge Set*, wenn G nach Entfernen aller Kanten $x \in X$ azyklisch ist. Gib einen effizienten Algorithmus an, der in G ein Feedback Edge Set mit minimalem Gewicht findet. Begründe außerdem die Korrektheit des Algorithmus und analysiere seine Laufzeit.
Hinweis: die beiden vorherigen Teilaufgaben sind hierfür hilfreich. (3 Punkte)

Aufgabe 2 - Darf's ein bisschen weniger sein? (3 Punkte)

Gegeben sei ein zusammenhängender, gewichteter Graph $G = (V, E)$ und ein Knoten $s \in V$. Gib einen Algorithmus an, der für jeden Knoten $t \in V$ das minimale Kantengewicht w_t bestimmt, sodass es einen s-t-Pfad gibt, bei der jede Kante ein Gewicht von höchstens w_t hat. Begründe die Korrektheit deines Algorithmus. Nenne und begründe außerdem seine asymptotische Laufzeit.

Aufgabe 3 - Amortisierte Analyse (7 Punkte)

Wir betrachten eine Datenstruktur, die zur Verwaltung von Elementen genutzt wird. Dabei wird zwischen roten und blauen Elementen unterschieden. Wir bezeichnen mit n_b die Anzahl blauer und mit n_r die Anzahl roter Elemente in der Datenstruktur. Es stehen die folgenden Operationen zur Verfügung:

- $\textsc{add}()$: Einfügen eines blauen Elements in $\Theta(1)$
- $\textsc{mark}()$: Umfärben der aufgerundeten Hälfte der blauen Elemente auf rot in $\Theta(n_b)$
- $\textsc{remove}()$: Löschen aller roten Elemente in $\Theta(n_r)$

1. Gib für die folgenden beiden Abfolgen von Operationen an die Gesamtkosten an:

$$k \cdot \textsc{add}, \textsc{mark}, \textsc{remove}$$

$$m \cdot (k \cdot \textsc{add}, \textsc{mark})$$

(2 Punkte)

2. Zeige mit Hilfe der Kontomethode, dass in jeder beliebigen Abfolge von Operationen jede Operation amortisiert konstante Kosten hat. (2 Punkte)

3. Zeige mit Hilfe der Potentialmethode, dass in jeder beliebigen Abfolge von Operationen jede Operation amortisiert konstante Kosten hat. (3 Punkte)