Algorithmen und Datenstrukturen

# Übungsblatt 6

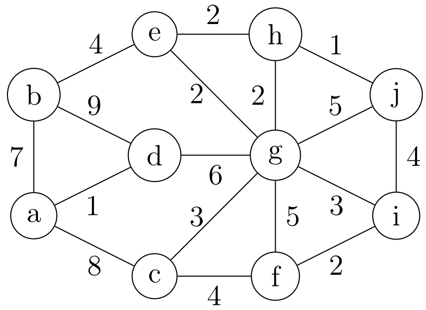
Prof. Dr. U. Schöning
T. Büchler
Institut für Theoretische Informatik

Wintersemester 2020/21

Abgabe bis 18.12. **13:00 Uhr**, Besprechung: 21.12.20-8.01.21

**Aufgabe 6.1** *4 Punkte.*

Berechnen Sie den minimalen aufspannenden Baum des nebenstehenden Graphen mithilfe des Kruskal-Algorithmus. Geben Sie die Reihenfolge an, in der die Kanten ausgewählt werden. Welches Gewicht hat der minimal aufspannende Baum?




**Aufgabe 6.2** *3 Punkte.*

Entwickeln Sie einen Greedy-Sortieralgorithmus mit Laufzeit $\Theta(n^2)$. Geben Sie den Algorithmus in Pseudocode an. Eingabe ist eine unsortiertes Array $A$ der Länge $n$.

**Aufgabe 6.3** *4 Punkte.*

Sei $G = (V, E)$ ein gerichteter Graph mit $|V| = n$ Knoten und $|E| = \mathcal{O}(n \log n)$ Kanten. Der Graph kann mittels einer Adjazenzliste oder als Adjazenzmatrix gespeichert werden. Geben Sie für beide Datenstrukturen die Zeit in $\mathcal{O}$-Notation an, die folgende Operationen im Erwartungswert benötigen:

(a) Prüfen ob $(u, v) \in E$

(b) Für einen Knoten $u$ alle Kanten $(u, v) \in E$ auszugeben.

(c) Für einen Knoten $v$ alle Kanten $(u, v) \in E$ auszugeben.

(d) Welche Zeiten ändern sich, wenn der Graph $|E| = \mathcal{O}(n^2)$ Kanten hat?

**Aufgabe 6.4** *5 Punkte.*

In der Vorlesung haben Sie die Darstellung der Union-Find-Datenstruktur als Array aus Zeigern kennengelernt. Gegeben sei folgendes Array:

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 2 | 4 | 4 | 4 | 6 | 8 | 8 | 8 | 10 | 12 | 12 | 12 | 14 | 16 | 16 | 16 |

(a) Führen Sie die Operationen union(1,5), union(11,13) und union(1,10) nacheinander aus und geben Sie nach jedem Schritt das entstandene Array an.

(b) Führen Sie anschließend find(2) zuerst ohne, dann mit Pfadverkürzung aus. Wie verändert sich jeweils das Array?

– (Beachten Sie die nächste Seite) –

**Aufgabe 6.5** *4 Punkte.*

Zwei Gauner erbeuten bei einem Raubzug eine Kassette mit Diamanten. Als die Gauner die Kassette inspizieren, stellen sie entsetzt fest, dass es sich um eine Mustersammlung eines Diamantenherstellers handelt, das heißt, alle Diamanten sind unterschiedlich groß. Da die Gangster die genaue Umrechnung zwischen Gewicht und Wert nicht kennen, und die Gewichte streng monoton zunehmen, entscheiden sie sich, die Diamanten gemäß folgender Strategie aufzuteilen:

Die Steine werden aufsteigend sortiert und anschließend wird wie folgt aufgeteilt:

- Gauner 1 nimmt den größten Stein
- Gauner 2 nimmt die nächsten zwei größten Steine
- Gauner 1 nimmt die nächsten zwei größten Steine
- $\cdots$
- Gauner 1 nimmt den letzten Stein

Sei hierbei $n$ die Anzahl der Steine. Sie dürfen annehmen, dass $n$ durch 4 teilbar ist. Bestimmen Sie eine streng monoton steigende Wertefunktion $w : \{1, \ldots, n\} \to \mathbb{N}$, die dem $i$-t kleinsten Diamanten den Wert $w(i)$ zuordnet, sodass

(a) Die Strategie der Gauner nicht optimal ist (ungleichmäßige Verteilung).

(b) Die Strategie der Gauner optimal ist (gleichmäßige Verteilung).

Begründen Sie, warum die Aufteilung unter der genannten Strategie optimal/nicht optimal ist.

*Beachten Sie:* Die Wertefunktion darf nicht als rekursive Funktion definiert werden, sondern muss in expliziter Form genannt werden.

(Diese Woche gibt es keine Programmieraufgabe)