Prof. Dr. Nicole Megow
Moritz Buchem, Alexander Lindermayr, Bart Zondervan

Wintersemester 2024/2025

Algorithmentheorie

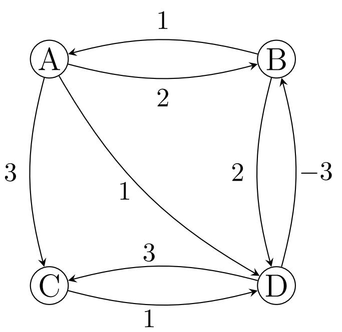
Übungsblatt 5 (Abgabe am 06.01.2025, 23:59 Uhr)

Übung 5.1 (3 Punkte)

Gebt für folgende Rekursionsgleichung eine geschlossene Form in Θ-Notation an und zeigt die Korrektheit per Induktion. Ihr könnt annehmen, dass $T(1) = 1$.

$$T(n) = T\left(\frac{n}{2}\right) + 3.$$

Übung 5.2 (5 Punkte)



Führt den Bellman-Ford-Algorithmus auf obigem Graphen mit Startknoten A aus. Die Kanten werden dabei in lexikographischer Reihenfolge betrachtet, d.h. $(A, B), (A, C), (A, D), (B, A), (B, D), (C, D), (D, B), (D, C)$. Ihr dürft den Algorithmus vorzeitig abbrechen, wenn sich keine weiteren Änderungen mehr ergeben. Gebt für jede Runde des Algorithmus die Updates der Distanzlabel an.

Übung 5.3 $(2 + 3 + 1$ Punkte$)$

Ihr seid Teamleiter bei einem IT-Dienstleister. Jede Woche kann Euer Team genau ein Aufgabenpaket erledigen. Ihr unterscheidet dabei zwischen Routineaufgaben, die ohne große Vorbereitung erledigt werden können, und komplexen Aufgaben. Eure Aufgabe ist es, für alle Wochen $1, \dots, n$ zu entscheiden, ob Ihr das Routine-Aufgabenpaket, das komplexe Aufgabenpaket, oder gar kein Aufgabenpaket bearbeitet. Durch das Bearbeiten des Routine-Aufgabenpakets in Woche $i \in \{1, \dots, n\}$ erzielt ihr Einnahmen in Höhe r_i. Durch das Bearbeiten des komplexen Aufgabenpakets in Woche $i \in \{1, \dots, n\}$ erzielt ihr Einnahmen in Höhe c_i. Das komplexe Aufgabenpaket benötigt allerdings eine intensive Vorbereitung, so dass Ihr, wenn Ihr in Woche i das komplexe Aufgabenpaket bearbeitet, in Woche $i - 1$ kein Aufgabenpaket bearbeiten könnt. Das bedeutet auch, dass Ihr in Woche 1 kein komplexes Aufgabenpaket bearbeiten könnt. Wenn Ihr in einer Woche kein Aufgabenpaket bearbeitet, erzielt ihr in dieser Woche keine Einnahmen. Ihr kennt also die Werte $r_1, \dots, r_n$ sowie $c_1, \dots, c_n$ und Eure Aufgabe ist es, die Aufgabenpakete für die Wochen $1, \dots, n$ so auszuwählen, dass maximale Einnahmen erzielt werden.

(a) Zeigt, dass der folgende Algorithmus nicht optimal ist, indem Ihr eine Instanz angebt, für die der Algorithmus eine falsche Antwort zurückgibt. Gebt für diese Instanz die optimale Lösung an sowie die Lösung, welche der Algorithmus zurückgibt. Der Algorithmus nimmt an, dass $r_i = c_i = 0$ für alle $i > n$.

Algorithmus 1 : Greedy Algorithmus

```
i := 1;
while i ≤ n do
    if c_{i+1} > r_i + r_{i+1} then
        print("Kein Aufgabenpaket in Woche i");
        print("Komplexes Aufgabenpaket in Woche i+1");
        i := i + 2;
    else
        print("Routine-Aufgabenpaket in Woche i");
        i := i + 1;
```

(b) Gebt einen Algorithmus an, der in Polynomialzeit mit Hilfe von dynamischer Programmierung die maximal erzielbaren Einnahmen berechnet. Gebt insbesondere auch die induktive Definition der DP-Tabelle an. Begründet, warum Euer Algorithmus korrekt ist und die gewünschte Laufzeit erreicht.

(c) Erklärt, wie aus der DP-Tabelle Eures Algorithmus der Aufgabenplan berechnet werden kann, welcher das maximale Einkommen erzielt.

Übung 5.4 $(2 + 1 + 2 + 1$ Punkte$)$

Sei G ein zusammenhängender, gewichteter, ungerichteter Graph mit $|V(G)| \geq 2$ und Gewichtsfunktion $c\colon E(G) \to \mathbb{R}_{>0}$. Zeigt oder widerlegt folgende Aussagen.

(a) Für jeden MST T von G gibt es zwei verschiedene Knoten $s, t \in V(G)$ so dass es einen kürzesten Weg in G zwischen s und t gibt, welcher nur Kanten aus T verwendet.

(b) Für jeden MST T von G gibt es einen Knoten $s \in V(G)$ so dass es für jeden Knoten $t \in V(G)$ mit $s \neq t$ einen kürzesten Weg in G zwischen s und t gibt, welcher nur Kanten aus T verwendet.

(c) Es gibt einen MST T von G und einen Knoten $s \in V(G)$ so dass es für jeden Knoten $t \in V(G)$ mit $s \neq t$ einen kürzesten Weg in G zwischen s und t gibt, welcher nur Kanten aus T verwendet.

(d) Alle Kantenkosten in G sind gleich 1 und es gibt einen MST T von G und einen Knoten $s \in V(G)$ so dass es für jeden Knoten $t \in V(G)$ mit $s \neq t$ einen kürzesten Weg in G zwischen s und t gibt, welcher nur Kanten aus T verwendet.