Approximationsalgorithmen

Wintersemester 2019/2020

Dr. Annamaria Kovacs
Dipl-Math. Mahyar Behdju

GOETHE UNIVERSITÄT FRANKFURT AM MAIN
Institut für Informatik
AG Algorithm Engineering

Übung 4

Ausgabe: 06.11.2019
Abgabe: 13.11.2019

Aufgabe 4.1. Dynamische Programmierung (2 + 1 + 3 Punkte)

Bestimme nur die geeigneten Rekursionsgleichungen für die folgenden (Teil-)Probleme.

a) Wir beschränken das Rucksack-Problem auf Eingaben mit $w_1, \dots, w_n \in \mathbb{N}$, $g_1, \dots, g_n, G \in \mathbb{R}^+$. Sei $W = \sum_{i=1}^{n} w_i$. Für $w \in \{0, 1, 2, \dots, W\}$ bezeichne $G(i, w)$ das minimale Gesamtgewicht einer Auswahl aus den ersten i Objekten mit Gesamtwert genau w.

$$G(0, w) = \dots$$
$$G(i, w) = \dots$$

b) Wie findet man in Teil a) den optimalen *Wert* für die gesamte Instanz?

c) Ein Gehweg soll mit einem taktilen Leitstreifen der Breite 20 cm aus Spezial-Steinplatten versehen werden. Die verfügbaren Platten haben Breite 20 cm und Länge 50, 60 oder 80 cm. Bezeichne $M(l)$ die minimale *Anzahl* von Steinplatten, mit denen ein Streifen der Länge $l \cdot 10$ cm gebaut werden kann, bzw. sei $M(l) = \infty$, falls dies mit den vorhandenen Plattenlängen nicht machbar ist.

$$M(0) = \dots$$
$$M(l) = \dots$$

Aufgabe 4.2. *Dynamische Programmierung auf Bäumen* (4 + 2 + 3 Punkte)

Im Problem max-Gewicht-Independent-Set wird in einem Eingabegraphen $G = (V, E)$ mit Knotengewichten w_v (für $v \in V$) eine unabhängige Knotenmenge $I \subseteq V$ mit maximalem Gesamtgewicht der Knoten in I gesucht.

Entwirf einen effizienten Algorithmus mit dynamischer Programmierung für max-Gewicht-Independent-Set für den Spezialfall, dass der Eingabegraph ein Baum ist.

a) Bestimme, wie das maximale Gesamtgewicht einer unabhängigen Knotenmenge *bottom-up* gefunden wird.

b) Bestimme, wie die entsprechende unabhängige Knotenmenge *top-down* gefunden wird.

c) Sei $G = (V, E)$ der einfache Weg $(v_0\, v_1\, v_2\, v_3\, v_4\, v_5)$ und sei v_0 die Wurzel von G. Bestimme Knotengewichte w_i für G, sodass es vom Gewicht von v_0 abhängt, ob v_5 in einer unabhängigen Knotenmenge mit maximalem Gewicht enthalten ist.

Bitte wenden!

Aufgabe 4.3. (1 + 3 + 1 Punkte)

Für jedes Dreieck (A, B, C) in der Ebene sei ein Gewicht $W(A, B, C)$ definiert, das einfach zu berechnen ist (z. B. Umfang, Länge der längsten Seite, Größe des größten Winkels oder Ähnliches).

Wir wollen für ein beliebiges konvexes Polygon, gegeben durch seine im Uhrzeigersinn aufgezählten n Eckpunkte $(P_1, P_2, P_3, \ldots, P_n)$, $n \geqslant 3$, eine Triangulierung (Zerlegung in Dreiecke) mit minimalem Gesamtgewicht der Dreiecke berechnen. Gesucht wird ein Algorithmus mit dynamischer Programmierung, der dieses Problem optimal löst.

Für $1 \leqslant i < i + 2 \leqslant j \leqslant n$ bezeichne $L(P_i, P_j)$ das optimale Gesamtgewicht einer Triangulierung des Teilpolygons $(P_i, P_{i+1}, P_{i+2}, \ldots, P_j)$.

a) Bestimme $L(P_i, P_{i+2})$ für $i = 1, 2, \ldots, n{-}2$.

b) Gib eine Rekursionsgleichung für $L(P_i, P_j)$ für $j \geq i{+}3$ an.

c) Welches Teilproblem $L(P_i, P_j)$ ergibt letztendlich den optimalen Wert für das gesamte Polygon?

E-Mail: `{panni,mahyar}@ae.cs.uni-frankfurt.de`