Prof. Dr. Martin Hoefer
Tim Koglin, Marco Schmalhofer

GOETHE UNIVERSITÄT FRANKFURT AM MAIN
Institut für Informatik
Algorithmen und Komplexität

Übungsblatt 11

Ausgabe: 28.06.2022
Abgabe: 05.07.2022, **08:00**

Dies ist das letzte Übungsblatt der Veranstaltung.
Die letzten Tutorien finden in der Woche vom 11.7. bis zum 15.7. statt.

Aufgabe 11.1 *Huffman-Codes* (10 Punkte)

Bestimmen Sie die Häufigkeit der vorkommenden Zeichen, den Huffman-Baum und die resultierende Kodierungstabelle sowie den Huffman-Code für den folgenden Ausdruck:

AUF DEM RASEN RASEN HASEN

Gehen Sie dabei wie folgt vor:
Als linkes Kind soll stets der Buchstabe mit kleinerem Häufigkeitswert eingefügt werden, bei gleichem Häufigkeitswert der alphabetisch kleinere. Bei Zusammenfassung mehrerer Buchstaben wird die gesamte Menge durch den kleinsten Buchstaben (bezüglich des Alphabets) repräsentiert, der in der Menge vorkommt. Buchstaben, die gleich häufig vorkommen, sollen (aufsteigend) nach alphabetischer Reihenfolge behandelt werden, wobei das Leerzeichen in dieser Sortierung größer als alle Buchstaben sein soll.

Aufgabe 11.2 *Auch Haustiere haben Präferenzen* (3 + 3 Punkte)

Wir betrachten in dieser Aufgabe das Stabile Matching-Problem mit *drei* Parteien (3PSM). Gegeben ist eine Menge A von Frauen, eine Menge B von Männern sowie eine Menge C von Haustieren, wobei $|A| = |B| = |C| = n$. Jede Frau $a \in A$ hat eine Präferenzordnung $\succ_a$ über alle Paare in $B \times C$. Ebenso hat jeder Mann $b \in B$ eine Präferenzordnung $\succ_b$ über alle Paare in $A \times C$, und jedes Haustier $c \in C$ eine Präferenzordnung $\succ_c$ über alle Paare in $A \times B$. Gesucht ist ein stabiles Matching über den drei Parteien.

Ein Matching ist eine Menge $M \subseteq A \times B \times C$ von Tripeln, so dass jede Frau $a \in A$, jeder Mann $b \in B$ und jedes Haustier $c \in C$ maximal in einem der Tripel in M vorkommt.

Ein Matching M heißt *stabil*, wenn es kein blockierendes Tripel $(a, b, c) \in A \times B \times C$ gibt. Ein Tripel (a, b, c) heißt *blockierend*, wenn Frau a, Mann b und Haustier c alle drei lieber dieses Tripel bilden als in ihrem aktuellen Tripel in M (oder ungematcht) zu bleiben.

Gegeben sei folgende Instanz des Problems: $A{=}\{1, 2\}$, $B{=}\{3, 4\}$, $C{=}\{5, 6\}$, mit Präferenzen

$$\begin{aligned}
&\succ_1: \big((3,5),(3,6),(4,6),(4,5)\big), && \succ_2: \big((3,5),(4,6),(3,6),(4,5)\big),\\
&\succ_3: \big((2,6),(2,5),(1,6),(1,5)\big), && \succ_4: \big((1,6),(2,6),(2,5),(1,5)\big),\\
&\succ_5: \big((2,4),(2,3),(1,3),(1,4)\big), && \succ_6: \big((1,3),(2,4),(1,4),(2,3)\big).
\end{aligned}$$

a) Geben Sie ein blockierendes Tripel für das Matching $M = \{(1, 3, 5), (2, 4, 6)\}$ an.

b) Geben Sie ein stabiles Matching an.

Aufgabe 11.3 *Übungsblätter erstellen* (2 + 4 + 1 + 2 Punkte)

Zwei Übungsleiter haben eine Menge B von n Übungsaufgaben zur Verfügung, die sie vollständig für die zwei kommenden Übungsblätter verwenden möchten. Jede Übungsaufgabe $i \in \{1, \ldots, n\}$ hat einen Schwierigkeitsgrad $d_i \in \mathbb{N}_{>0}$. Der Schwierigkeitsgrad eines Übungsblatts entspricht der Summe der Schwierigkeitsgrade der darauf gestellten Übungsaufgaben. Ferner sei $D = \sum_{i \in B} d_i$ der kumulierte Schwierigkeitsgrad aller Übungsaufgaben in B.

Entwerfen und analysieren Sie ein dynamisches Programm, das entscheidet, ob es möglich ist, alle n Übungsaufgaben in B so auf die zwei kommenden Übungsblätter zu verteilen, dass beide den gleichen Schwierigkeitsgrad haben. Gehen Sie dabei wie folgt vor:

a) Definieren Sie geeignete Teilprobleme T.

   Hinweis: Es genügt, die Teilprobleme T so zu definieren, dass sie binär, d.h., entweder mit 1 ("ist wahr") oder mit 0 ("ist falsch"), beantwortet werden können.

b) Geben Sie Basisfälle sowie die Rekursionsgleichung für die Teilprobleme an.

c) Durch den Aufruf welches Teilproblems kann nun entschieden werden, ob die zwei kommenden Übungsblätter den gleichen Schwierigkeitsgrad haben können?

d) Beschreiben Sie, wie Ihr dynamisches Programm implementiert werden kann (kein Pseudocode erforderlich) und geben Sie die resultierende Laufzeit in Abhängigkeit von n und D an.

Aufgabe 11.4 *Dynamischer Taxifahrer* (2 + 4 + 2 + 3 Punkte)

Ein Taxifahrer erledigt Fahrten zwischen den verschiedenen Taxiständen $1, \ldots, k$. Die Fahrzeit in Minuten von Stand u zu Stand v ist gegeben durch $\mathsf{zeit}(u, v) \in \mathbb{N}$, wobei $\mathsf{zeit}(u, u) = 0$ und $\mathsf{zeit}(u, v) > 0$ für $u \neq v$. Ein *Auftrag* ist ein 4-Tupel bestehend aus Abfahrtszeit, Startpunkt, Zielpunkt und Bezahlung. Für das Jahr 2023 wurden dem Taxifahrer vorab n Aufträge angeboten. Auftrag $i \in \{1, \ldots, n\}$ wird durch das 4-Tupel

$$(a_i, s_i, z_i, b_i) \in \mathbb{N} \times \{1, \ldots, k\} \times \{1, \ldots, k\} \times \mathbb{R}_{\geq 0}$$

beschrieben. Dabei ist a_i die Abfahrtszeit in Minuten seit 1. Januar 2023, 00:00 Uhr, s_i der Startpunkt, z_i der Zielpunkt und b_i die Bezahlung.

Der Taxifahrer möchte herausfinden, welche Aufträge er annehmen soll, um seine Jahreseinnahmen zu maximieren. Die angenommenen Aufträge müssen folgende Bedingungen erfüllen:

(I) Keine zwei Aufträge dürfen sich zeitlich überlappen.

(II) Zwischen je zwei aufeinanderfolgenden Aufträgen muss genug Zeit sein, um vom Zielpunkt des ersten („leer“) zum Startpunkt des zweiten zu fahren.

Die Anzahl der Leerfahrten zwischen den Aufträgen und deren Kosten sind ihm egal. Ebenso ist es ihm egal, wo die erste Fahrt beginnt und wo die letzte aufhört.

Wir definieren für ein dynamisches Programm die Teilprobleme

$$\begin{aligned}\mathrm{opt}(t, v) = {} & \text{maximal erreichbare Einnahmen bis zum Zeitpunkt } t\text{, wenn} \\ & \text{das Taxi zum Zeitpunkt } t \text{ an Stand } v \text{ sein muss}\end{aligned}$$

für $t \in \{0, \ldots, T\}$ und $v \in \{1, \ldots, k\}$. Dabei ist $T = \max_{1 \leq i \leq n}(a_i + \mathsf{zeit}(s_i, z_i))$ der späteste Zeitpunkt, an dem ein Auftrag endet.

a) Geben Sie geeignete Basisfälle für obige Teilprobleme an und erläutern Sie deren Bedeutung.

b) Geben Sie eine Rekursionsgleichung für obige Teilprobleme an. Erläutern Sie die Bedeutung der einzelnen Bestandteile der Rekursionsgleichung.

c) Beschreiben Sie, wie die maximal erreichbaren Jahreseinnahmen mithilfe der ausgefüllten Lösungstabelle berechnet werden können.

d) Beschreiben Sie, wie mithilfe von a), b) und c) ein Algorithmus implementiert werden kann, der die maximal erreichbaren Jahreseinnahmen in Zeit $\mathcal{O}(T \cdot n)$ berechnet, wenn k eine Konstante ist (kein Pseudocode erforderlich).