Karlsruher Institut für Technologie

Institut für Theoretische Informatik
T.T.-Prof. Dr. Thomas Bläsius
https://scale.iti.kit.edu

Übungsblatt 07

Algorithmen I – Sommersemester 2022

Abgabe im ILIAS bis 15.06.2022, 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 - Fahrplanauskunft (7 Punkte)

Eine elementare Zugverbindung C ist ein Tupel $C = (Z, B_{\text{ab}}, B_{\text{an}}, \tau_{\text{ab}}, \tau_{\text{an}})$ mit der Bedeutung, dass der Zug Z zum Zeitpunkt τ_{ab} am Bahnhof B_{ab} losfährt und ohne Zwischenstopp zum Zeitpunkt τ_{an} am Bahnhof B_{an} ankommt. Sei $\mathcal{C}$ eine Menge solcher elementarer Verbindungen und sei $\mathcal{B}$ die Menge aller Bahnhöfe. Eine Routenanfrage besteht aus einem Startbahnhof $B_s \in \mathcal{B}$, einem Zielbahnhof $B_t \in \mathcal{B}$, sowie einem Startzeitpunkt τ_s. Ziel ist es, eine Zugverbindung von B_s nach B_t zu finden, die nicht vor τ_s startet und möglichst früh endet.

1. Modelliere das Problem der Fahrplanauskunft als kürzeste Wege Problem in einem Graphen. Wende dann deine Modellierung auf die folgende Probleminstanz an und zeichne den resultierenden Graphen: (4 Punkte)

$$\mathcal{B} = \{\text{Growth}, \text{ Progress}, \text{ QualityCity}\}$$

$$
\begin{array}{llllll}
\mathcal{C} = \{ & (Z_0, & \text{Growth}, & \text{QualityCity}, & (09 \text{ Uhr}), & (12 \text{ Uhr})) \\
& (Z_1, & \text{Growth}, & \text{Progress}, & (12 \text{ Uhr}), & (13 \text{ Uhr})) \\
& (Z_2, & \text{QualityCity}, & \text{Growth}, & (12 \text{ Uhr}), & (15 \text{ Uhr})) \\
& (Z_3, & \text{Progress}, & \text{QualityCity}, & (13 \text{ Uhr}), & (17 \text{ Uhr})) \\
& (Z_4, & \text{Progress}, & \text{Growth}, & (11 \text{ Uhr}), & (12 \text{ Uhr})) \\
& \} & & & &
\end{array}
$$

Hinweis: Du darfst annehmen, dass mit Uhrzeiten gerechnet werden kann wie mit ganzen Zahlen.

2. Sei nun für jeden Bahnhof noch eine Umsteigezeit gegeben und für je zwei Bahnhöfe B_1 und B_2 die Dauer, in der man von B_1 nach B_2 laufen kann. Erweitere deine Modellierung so, dass die Umsteigezeiten eingehalten werden und die Fußwege zwischen Bahnhöfen beachtet werden. (3 Punkte)

Aufgabe 2 - Dijkstra und negative Kanten (2 Punkte)

In dieser Aufgabe wollen wir untersuchen, was passiert wenn Dijkstra's Algorithmus auf einem gerichteten gewichteten Graphen mit (unter anderem) negativen Kantengewichten aufgerufen wird. Wir beschränken uns hierbei auf Instanzen ohne negative Kreise (d.h. die Summe der Kantengewichte in jedem Kreis ist nicht negativ). Gib einen Graphen G und zwei Knoten s und t an, sodass Dijkstra's Algorithmus aus der Vorlesung auf (G, s) nicht den kürzesten s-t-Weg findet und begründe kurz weshalb. (2 Punkte)

Aufgabe 3 - Das Orakel von Dijkstra (6 Punkte)

Zu einem Graphen $G = (V, E)$ und einem Knoten $s \in V$ bezeichnen wir mit dem *Kürzeste-Wege-Baum* von s einen Baum $T = (V, E')$ mit $E' \subseteq E$ bei dem jeder Pfad von s nach $t \in V$ in T auch einem kürzesten Pfad von s nach t in G entspricht. Entscheide für jede der folgenden Aussagen, ob sie wahr oder falsch ist und begründe deine Antwort.

1. Wenn alle Kanten im (gerichteten) Graph paarweise verschiedene Gewichte haben, dann ist der Kürzeste-Wege Baum von jedem Startknoten s aus eindeutig. (1 Punkt)

2. Wenn wir die Kantenrichtungen in einem gerichteten Graphen mit positiven Kantengewichten entfernen (indem man jede Kante als ungerichtet auffasst), dann ändern sich die Distanzen (bezüglich kürzester Wege) nicht. (1 Punkt)

3. Wenn wir das Gewicht jeder Kante um $k > 0$ vergrößern, dann vergrößern sich die Distanzen zwischen Knotenpaaren um Vielfache von k. (1 Punkt)

4. Wenn wir das Gewicht jeder Kante um $k > 0$ verkleinern, dann verkleinern sich die Distanzen zwischen Knotenpaaren um Vielfache von k. (1 Punkt)

5. Unter allen kürzesten Wegen zwischen zwei Knoten in einem (gerichteten) gewichteten Graph findet Dijkstra's Algorithmus immer denjenigen mit den wenigsten Kanten. (1 Punkt)

6. Falls ein zusammenhängender Graph keine negativen Kreise enthält, dann gibt es zwischen jedem Knotenpaar einen kürzesten Pfad, der einfach ist. Zur Erinnerung: ein Pfad ist einfach, wenn er keine Kreise enthält. (1 Punkt)

Aufgabe 4 - Viele Wege führen nach Rom (5 Punkte)

1. Gibt es Graphen mit n Knoten, in denen es zwischen zwei Knoten $2^{\Theta(n)}$ kürzeste Wege gibt? Begründe deine Antwort. (2 Punkte)

   Hinweis: Um eine asymptotische untere Schranke zu zeigen genügt es nicht einen einzelnen Graphen anzugeben. Versuche stattdessen eine Konstruktion zu finden, die für beliebig große n funktioniert.

2. Gegeben sei ein ungewichteter Graph $G = (V, E)$ mit $|V| = n$ und $|E| = m$. Beschreibe einen Algorithmus der in $O(n + m)$ zu einem gegebenen $s \in V$ für alle $v \in V$ die Anzahl kürzester s-v-Wege bestimmt. Begründe die Korrektheit deines Algorithmus und wieso er das geforderte Laufzeitverhalten hat. (3 Punkte)

Dr. Meta hat sich bereits an einem geheimen Urlaubsort zurückgezogen, um sich von dem beruflichen Stress der letzten Wochen zu erholen.
Er wünscht eine erholsame Pfingst-Pause!