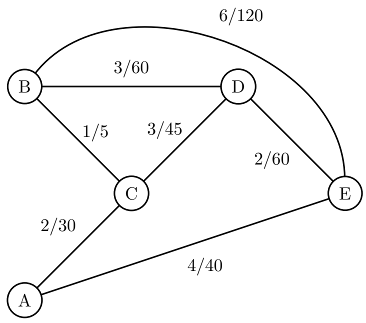
Übungsblatt 6

Algorithmen und Datenstrukturen (WS 2013, Ulrike von Luxburg)

Präsenzaufgabe 1 (Dijkstra-Algorithmus) Betrachten Sie folgende “Straßenkarte”:



Jeder Straßenabschnitt ist darin mit zwei Werten ℓ/s gelabelt: Seiner Länge ℓ in km und der darauf erreichbaren Durchschnittsgeschwindigkeit s in km/h.

(a) Bestimmen Sie alle von A ausgehenden *kürzesten* Wege durch Anwendung des Djikstra-Algorithmus (mit Min-Priority-Queue, Folie ≈ 255). Geben Sie zum Zeitpunkt jeder Ausführung von Zeile 3 tabellarisch die Werte $v.dist$ und $v.\pi$ für alle Knoten v an.

(b) Bestimmen Sie ebenso alle von A ausgehenden *schnellsten* Wege (in Minuten).

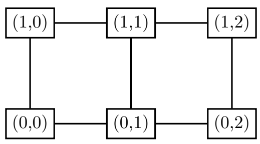
(c) Wie lautet jeweils der kürzeste/schnellste Weg von A nach B?

Präsenzaufgabe 2 (Dijkstra trotz negativer Kantengewichte) Sei $G = (V, E)$ ein gerichteter Graph mit Gewichtsfunktion $w : V \times V \to \mathbb{R}$ ohne negative Zyklen. Wir haben gesehen, dass der Dijkstra-Algorithmus selbst aufgrund einzelner negativer Kantengewichte fehlschlagen kann. Im Folgenden sind daher zwei Vorschläge, um den Dijkstra-Algorithmus auf einem Graphen $G' = (V, E)$ mit modifizierten nicht-negativen Gewichten w' anzuwenden. Die Behauptung ist bei beiden Vorschlägen, dass für alle $s, t \in V$ der kürzeste Pfad von s nach t in G' auch ein kürzester Pfad (ggf. anderer Länge) von s nach t in G ist. Beweisen oder widerlegen Sie jeweils.

(a) Setze $w'_{ij} := w_{ij} - m$ für $m := \min_{i,j \in V} w_{i,j}$.

(b) Sei $h : V \to \mathbb{R}$ beliebig so, dass $w'_{ij} := w_{ij} + h(i) - h(j) \geq 0$ für alle $i, j \in V$.

Präsenzaufgabe 3 (Gittergraph) Sei $G = (V, E)$ der zweidimensionale $m \times n$-Gitter-Graph, definiert per $V = \{0, \ldots, m-1\} \times \{0, \ldots, n-1\}$ und $E = \{((a, b), (c, d)) \in V \times V \mid |a-c| + |b-d| = 1\}$. Zur Visualisierung wird der Knoten (x, y) auf der entsprechenden kartesischen Koordinate platziert, siehe folgendes Beispiel für $m = 3$ und $n = 2$:



(a) In welcher (Kürzesten-Wege-)Distanz liegt die untere linke Ecke zur oberen rechten?

(b) In welcher Distanz liegen zwei beliebige Knoten $(a, b) \in V$ und $(c, d) \in V$?

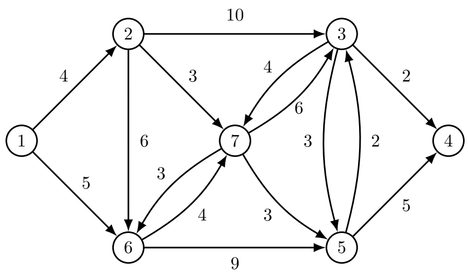
(c) *Wieviele* verschiedene kürzeste Wege gibt es zwischen $(a, b) \in V$ und $(c, d) \in V$?

Begründen Sie jeweils kurz.

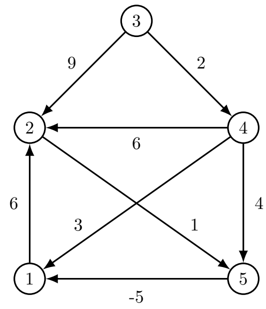
Hausaufgaben zum 15. Januar, 9:00 (Vorlesungsbeginn).

Aufgabe 1 (Dijkstra-Algorithmus (3+3)) Gegeben sind folgende zwei Graphen:

$G_1 :=$



$G_2 :=$



(a) Bestimmen Sie in G_1 alle von Knoten 1 ausgehenden kürzesten Pfade, indem Sie den Dijkstra Algorithmus (mit Min-Priority-Queue) anwenden. Geben Sie zum Zeitpunkt jeder Ausführung von Zeile 3 tabellarisch die Werte $v.dist$ und $v.\pi$ für alle Knoten v an.
Lesen Sie daran den kürzesten Pfad von 1 nach 4 ab.

(b) Belegen Sie, dass der Dijkstra-Algorithmus (mit Min-Priority-Queue) in G_2 für das Single-Source-Shortest-Path Problem zu einem falschen Ergebnis führen kann.

Aufgabe 2 (Dijkstra-Modifikation (4)) Sei $G = (V, E)$ gerichtet mit nicht-negativen Kantengewichten und $s \in V$. Modifizieren Sie den Dijkstra-Algorithmus (mit Min-Priority-Queue) so, dass er von s zu jedem anderen Knoten einen Pfad liefert, dessen schwerste Kante so leicht wie möglich ist. Begründen Sie kurz Ihre Modifikation.

Aufgabe 3 (Adjazenzmatrix und Pfade (3+2+2)) Sei A die $n \times n$-Adjazenzmatrix eines beliebigen ungewichteten Graphen $G = (V, E)$, und bezeichne $A[i, j]$ den Eintrag in der i-ten Zeile und j-ten Spalte.

(a) Beweisen Sie (per Induktion) für alle $k \in \mathbb{N}_{\geq 0}$, dass $A^k[i, j]$ der Anzahl verschiedener Pfade der Länge k entspricht, die in G von i nach j führen.

(b) Nutzen Sie (a) für einen Algorithmus, der für alle Knotenpaare gleichzeitig ermittelt, ob diese jeweils mit einem Pfad der Länge *genau* k für ein $k \geq 0$ verbunden werden können.

(c) Modifizieren Sie die Eingabematrix ihres Algorithmus aus (b) so, dass er ermittelt, ob die jeweiligen Knotenpaare mit einem Pfad der Länge *höchstens* k verbunden werden können.

Aufgabe 4 (Kürzeste Pfade in Routingnetzwerken (2+2)) Sei N ein zusammenhängender, ungerichteter, ungewichteter Graph. Wir fassen N als "Netzwerk" auf und wollen von jedem Knoten u zu jedem Knoten v ein Datenpaket entlang eines einfachen Pfades $p_{u,v}$ senden. Wir definieren dazu ein Pfadsystem $W := \{p_{u,v} \mid (u, v) \in V \times V\}$. Dies impliziert für jede Kante e die Kantenlast $c(e) := |\{p_{u,v} \in W \mid e \in p_{u,v}\}|$ als die Anzahl der über $e \in E$ gerouteten Pakete. Ein wichtiger Parameter für die Güte eines Pfadsystems ist die maximale sich durch W ergebene Last, also $c(W) := \max_{e \in E} c(e)$. Sei nun $W^* = \{p_{u,v} \mid (u, v) \in V \times V$ und $p_{u,v}$ ist ein kürzester Pfad von u nach $v\}$ ein spezielles Pfadsytem, welches stets nur entlang irgendeines festen kürzesten Weges routet.

(a) Beweisen Sie: Für jedes Pfadsystem W gilt, dass $c(W) \geq \frac{1}{|E|} \sum_{p \in W} \ell(p)$, wobei $\ell(p)$ die Länge des Pfades p ist.

(b) Beweisen oder widerlegen Sie: $c(W) \geq c(W^*)$ für jedes Pfadsystem W, d.h., W^* ist optimal.