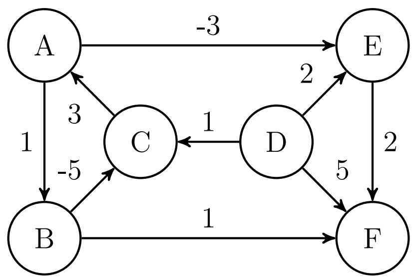
Algorithmen und Datenstrukturen II
SoSe 2024 – Serie 9

1 Bellmann-Ford

Gegeben sei der folgende gewichtete gerichtete Graph $G = (V, E, w)$. Beachten Sie die negativen Gewichte.



a) Benutzen sie den Bellman-Ford-Algorithmus, um in G die Längen der kürzesten Pfade von Knoten D zu allen anderen zu berechnen. In der inneren und äußeren Schleife werden die Kanten in lexikographischer Ordnung abgearbeitet, also (A, B), (A, E), (B, C), (B, F), (C, D), usw.

Geben sie nach jedem Durchlauf der inneren Schleife die jeweils bearbeitete Kante und die aktuellen Werte $D[i]$ für alle Knoten $i \in \{A, \ldots, F\}$ an. Beenden sie ihre Auflistung nach drei Durchläufen der äußeren Schleife.

Lösung:

Kante	*Gewicht*	$D[A]$	$D[B]$	$D[C]$	$D[D]$	$D[E]$	$D[F]$
(A,B)	1	∞	∞	∞	0	∞	∞
(A,E)	-3	∞	∞	∞	0	∞	∞
(B,C)	-5	∞	∞	∞	0	∞	∞
(B,F)	1	∞	∞	∞	0	∞	∞
(C,A)	3	∞	∞	∞	0	∞	∞
(D,C)	1	∞	∞	1	0	∞	∞
(D,E)	2	∞	∞	1	0	2	∞
(D,F)	5	∞	∞	1	0	2	5
(E,F)	2	∞	∞	1	0	2	4
(A,B)	1	∞	∞	1	0	2	4
(A,E)	-3	∞	∞	1	0	2	4
(B,C)	-5	∞	∞	1	0	2	4
(B,F)	1	∞	∞	1	0	2	4
(C,A)	3	4	∞	1	0	2	4
(D,C)	1	4	∞	1	0	2	4
(D,E)	2	4	∞	1	0	2	4
(D,F)	5	4	∞	1	0	2	4
(E,F)	2	4	∞	1	0	2	4
(A,B)	1	4	5	1	0	2	4
(A,E)	-3	4	5	1	0	1	4
(B,C)	-5	4	5	0	0	1	4
(B,F)	1	4	5	0	0	1	4
(C,A)	3	3	5	0	0	1	4
(D,C)	1	3	5	0	0	1	4
(D,E)	2	3	5	0	0	1	4
(D,F)	5	3	5	0	0	1	4
(E,F)	2	3	5	0	0	1	3

b) Nach wie vielen Iterationen terminiert der Algorithmus?

Lösung:

Während Bellman-Ford normal terminiert nach $|V|-1=5$ Durchläufen, kann der Algorithmus hier nicht die kürzere Wege erzeugen. Der negative Zyklus A-B-C-A kann beliebig oft durchlaufen werden, um immer kürzere Wege zu finden.

2 Postkutschenbeladung

Ein Beamter der königlichen Post von Makrokartonien belädt Postkutschen mit großen Paketen. Er beschliesst eine Kutsche so wertvoll wie möglich zu beladen mit den repetitiven Kunstschätzen seines Herren. Die Kutsche kann mit höchstens 600 kg beladen werden bevor die Kutschpferde rebellieren; Gewichte und Werte der Pakete sind in der folgenden Tabelle gegeben.

Pakettyp	Gewicht [kg]	Wert [Mio. Makrokartonesische Rupien]
A Antike Golddupplonen	150	70
B Unsortierte Edelsteine	250	210
C Silber-Barren	350	220
D Reichsäpfel	50	20

Jeder Pakettyp kann beliebig oft eingeladen werden (solange das Maximalgewicht nicht überschritten wird).
Lösen sie das Optimierungsproblem mit dem in der Vorlesung beschrieben dynamischen Programmierungs-Verfahren.

a) Geben Sie die Tabellen `best_wert` und `best_obj` für $j = 1, \ldots, 4$ an. Beachten Sie: nur wenige Indexwerte i müssen tabelliert werden; deshalb lassen sich die Tabellen relativ kompakt angeben.

b) Geben Sie die beste Beladung an.

c) Nach der Errechnung der Tabellen, verliert sich eines der Pferde in holistisch-metaphysischer Kontemplation der ADS-Übungsaufgaben *an sich* und sieht sich daher außerstande, weiterhin mit voller Kraft mitzuwirken. Die Kutsche kann deshalb nur noch mit 450 kg beladen werden. Was ist jetzt die beste Beladung?

Lösung:

a)

i/50	1	2	3	4	5	6	7	8	9	10	11	12
j=1												
best_wert/10	0	0	7	7	7	14	14	14	21	21	21	28
best_obj			A	A	A	A	A	A	A	A	A	A
j=2												
best_wert/10	0	0	7	7	21	21	21	28	28	42	42	42
best_obj			A	A	B	B	B	B	B	B	B	B
j=3												
best_wert/10	0	0	7	7	21	21	22	28	28	42	42	43
best_obj			A	A	B	B	C	B	B	B	B	C
j=4												
best_wert/10	2	4	7	9	21	23	25	28	30	42	44	46
best_obj	D	D	A	D	B	D	D	B	D	B	D	D

b) D,D,B,B

c) D,B,A

3 Hirschberg

Berechnen Sie das optimale Alignment für die beiden Strings $S_1 = ACBAAC$ und $S_2 = ABDABC$ mittels des Hirschberg-Algorithmus. Wenden Sie dabei die Rekursion auf S_1 an, dass heißt S_1 wird in jedem Schritt unterteilt. Hat der zu unterteilende String eine ungerade Länge, ist die Mitte dem linken Teilstring zuzuordnen. Gibt es mehrere optimale Score-Paare zur Fortsetzung des Algorithmus, so wählen Sie das erste Vorkommnis relativ zum Vorwärtsalignment. Geben Sie alle Schritte des Algorithmus an, bis jeder Teilstring die Länge 1 erreicht hat. In diesem Fall kennzeichnen Sie das resultierende Alignment in der Matrix. Geben Sie ebenfalls das finale Alignment beider Strings an.

Lösung:

Rekursionstiefe 1

	–	A	C	B					
–	0	1	2	3	3	4	5	6	A
A	1	0	1	2	3	3	4	5	B
B	2	1	1	$\boxed{1}$	$\boxed{2}$	2	3	4	D
D	3	2	2	2	1	1	2	3	A
A	4	3	3	3	2	1	1	2	B
B	5	4	4	3	2	1	0	1	C
C	6	5	4	4	3	2	1	0	–
					A	A	C	–	

Rekursionstiefe 2

	–	A	C			
–	0	1	2	1	2	A
A	1	0	$\boxed{1}$	$\boxed{0}$	1	B
B	2	1	1	1	0	–
				B	–	

	–	A	A			
–	0	1	2	3	4	D
D	1	1	2	2	3	A
A	2	1	$\boxed{1}$	$\boxed{1}$	2	B
B	3	2	2	0	1	C
C	4	3	3	1	0	–
				C	–	

Rekursionstiefe 3

	–	A			
–	0	1	1	1	A
A	1	$\boxed{0}$	$\boxed{1}$	0	–
			C	–	

	–	A			
–	0	1	1	2	D
D	1	$\boxed{1}$	$\boxed{0}$	1	A
A	2	1	1	0	–
			A	–	

Alignment:

```
ACBAA-C
A-BDABC
```