technische universität
dortmund

SoSe 2017

C. Sohler
A. Krivošija, A. Rey, H. Sandvoß

DAP2 – Heimübung 13

Ausgabedatum: 7. 7. 17 — Abgabedatum: Fr. 14. 7. 17 (Mo. 17. 7. für Gruppen 27–32) 12 Uhr

Schreiben Sie unbedingt immer Ihren **vollständigen Namen**, **Ihre Matrikelnummer** und **Ihre Gruppennummer** auf Ihre Abgaben!

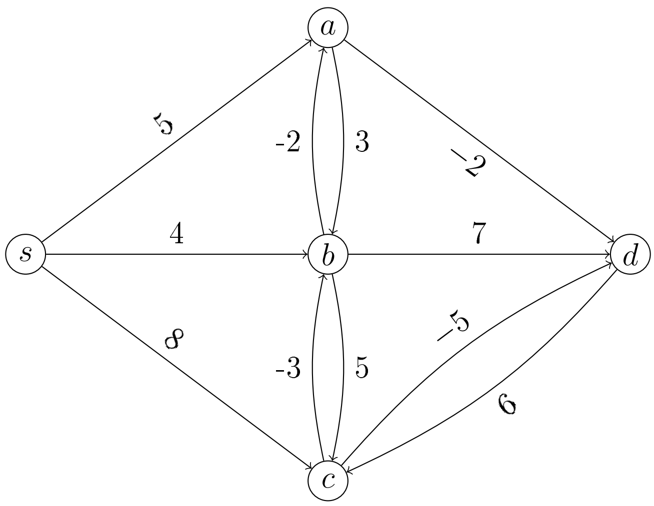
Aufgabe 13.1 (5 Punkte): (Negative Kanten und Bellman-Ford)

a) (2 Punkte) Bob schlägt für gerichtete Graphen $G = (V, E)$ mit Gewichtsfunktion w, die negative Kantengewichte, aber keine negativen Zyklen haben, die folgende Methode zur Berechnung kürzester Wege von einem Startknoten s vor:

- Wähle eine Konstante c so, dass für alle Kanten $e \in E$ gilt: $c + w(e) > 0$.
- Wähle als neue Gewichtsfunktion $w' : E \to \mathbb{R}$ mit $w'(e) = c + w(e)$.
- Berechne mit Hilfe des Dijkstra-Algorithmus die kürzesten Wege von s für den Graphen G mit der Gewichtsfunktion w'.

Warum stellt Bobs Methode kein korrektes Verfahren dar? Geben Sie ein Gegenbeispiel an und erläutern Sie dieses.

b) (3 Punkte) Führen Sie den Algorithmus von Bellman-Ford auf dem folgenden gerichteten Graphen $G = (V, E)$ aus. Als Startknoten dient der Knoten s. Geben Sie dabei nach der Initialisierung und nach jedem Durchlauf i der äußeren *for*-Schleife für jeden Knoten $v \in V$ den Wert $M[i][j]$ an. Ermitteln Sie mit Hilfe einer aus der Vorlesung bekannten Methode, ob der gegebene Graph G negative Zyklen enthält. Erläutern Sie die dazu nötigen zusätzlichen Schritte und Ihre Antwort gesondert.



Aufgabe 13.2 (5 Punkte): (Graphenalgorithmen)

Der Biologe Bob beobachtet und klassifiziert die Schmetterlinge, die einem der zwei Typen A und B gehört. Sei die Menge der beobachteten Schmetterlinge mit V bezeichnet. Dabei hat er seine Beobachtungen in zwei Mengen E und F gesammelt, so dass für jedes Paar der Schmetterlinge $(u, v) \in V \times V$ genau eine der folgenden Möglichkeiten gilt:

- das Paar (u, v) ist in E, was bedeutet dass u und v von gleichem Typ sind;
- das Paar (u, v) ist in F, was bedeutet dass u und v von unterschiedlichem Typ sind;
- das Paar (u, v) ist weder in E noch in F.

Jeder Schmetterling wurde mindestens einmal mit einem anderen Schmetterling verglichen (d. h. es wurde beobachtet, ob das Knotenpaar in E oder F liegt). Bob möchte jetzt wissen, ob seine Beobachtungen konsistent sind, d. h., ob es keine widersprüchlichen Aussagen über die Klassifizierung der Schmetterlinge in zwei Gruppen gibt.

a) Entwerfen Sie einen Algorithmus `Konsistent(`V, E, F`)`, der entscheidet, ob die Menge der Beobachtungen konsistent ist und dementsprechend `TRUE` bzw. `FALSE` zurückgibt. Beschreiben Sie den Algorithmus zunächst mit eigenen Worten. Setzen Sie den Algorithmus dann in Pseudocode um. Für die volle Punktzahl wird ein Algorithmus erwartet, dessen Laufzeit durch $\mathcal{O}(|V| + |E| + |F|)$ beschränkt ist.

b) Analysieren Sie die Laufzeit Ihres Algorithmus.

c) Beweisen Sie die Korrektheit Ihres Algorithmus.

Aufgabe 13.3 (5 Punkte): (**Bonusaufgabe:** Dynamische Programmierung)

Eine Laufbahn ist als ein zweidimensionales Feld mit $m \times n$ Blöcken dargestellt. In jedem Block des Feldes werden Strafpunkte angegeben: Im Block (i, j) sind $S(i, j)$ Strafpunkte für die Läufer anzurechnen, wobei $S(i, j) \geq 0$ gilt.

Emma möchte die Laufbahn durchlaufen, sodass sie die minimale Anzahl an Strafpunkten sammelt. Sie fängt in der ersten Spalte ($j = 1$) in einer beliebigen Zeile ($1 \leq i \leq m$) an und beendet ihren Lauf in der letzten Spalte ($j = n$). Die Regeln schreiben vor, dass sie von Block (i, j) ausgehend nur solche Blöcke betreten darf, die in der Spalte rechts von ihrem Block liegen und über eine Ecke oder über eine Seite an Block (i, j) angrenzen, d. h., die Blöcke, die oben rechts, rechts oder unten rechts von $(i.j)$ liegen.

Ein Beispiel ist unten gegeben. Von dem Block mit Punktewert 2 in Spalte $j = 1$ und Zeile 2 aus kann man nur die Werte 1 oder 2 (Zeilen 1, 2 oder 3 in Spalte 2) erreichen. Emma sollte die hervorgehobenen Blöcke besuchen, um so wenige Strafpunkte wie möglich zu sammeln. Die Pfeile kennzeichnen jeweils den Ein- und Ausgangsblock.

	$j=1$			$j=n$	
	8	1	0	3	
$\rightarrow$	$\boxed{\mathbf{2}}$	$\boxed{\mathbf{1}}$	7	3	
	4	2	$\boxed{\mathbf{1}}$	$\boxed{\mathbf{1}}$	$\rightarrow$
	3	3	1	4	

a) Geben Sie eine Rekursionsgleichung für die minimale Anzahl der Strafpunkte $E(i,j)$ an, die Emma in ihrem Lauf bis einschließlich Block (i,j) gesammelt haben muss.

b) Geben Sie einen Algorithmus in Pseudocode an, der auf dem Prinzip der dynamischen Programmierung beruht und die minimale Anzahl der von Emma gesammelten Punkte zurückgibt.

c) Analysieren Sie die Laufzeit Ihres Algorithmus.

d) Beweisen Sie die Korrektheit Ihrer Rekursionsgleichung aus Teilaufgabe a) mittels Induktion.