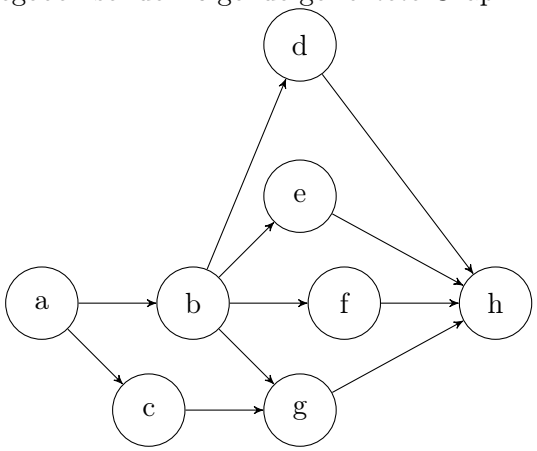
Algorithmen und Datenstrukturen II
SS 2018 – Serie 5

17 (6 Punkte) Pagerank

Gegeben sei der folgende gerichtete Graph



(a) Berechnen Sie den Pagerank des Graphen wie in der Vorlesung angegeben mit 1 Iteration. Ignorieren Sie bei der Berechnung das Eigengewicht des jeweiligen Knotens. Gehen Sie davon aus, dass die Knoten vor der ersten Iteration mit Pagerank 1 initialisiert wurden und der Normalisierungsfaktor c sowie der Dämpfungsfaktor d den Wert 1 haben.
Beginnen Sie ausgehend von Knoten a, ohne jedoch den Pagerank von a zu Beginn nochmal zu berechnen (Der Pagerank von a ist also nicht 0.).
Was stellen Sie fest, wenn Sie weitere Iterationen durchführen? (3 Punkte)

(b) Fügen Sie eine Kante {e,a} in den Graphen ein und wiederholen Sie Aufgabe a), wobei Sie die Berechnung bei Knoten a starten.
Was stellen Sie fest, wenn Sie weitere Iterationen durchführen?
Was stellen Sie fest, wenn Sie das Eigengewicht der jeweiligen Knoten – wie im verlinkten Paper beschrieben – mit in die Berechnung einbeziehen. (3 Punkte)

18 (3 Punkte) Editierdistanz

Gegeben sind die Zeichenketten

$U =$ LIPSIA
$V =$ LEIPZIG

Berechnen Sie die paarweise Editierdistanz zwischen U und V. Geben Sie die Matrix D_{ij} und ein optimales Alignment an.

19 (5 Punkte) Travelling Salesperson

Gegeben sei die folgende Entfernungsmatrix:

	1	2	3	4	5
1	0	5	7	9	3
2	2	0	1	3	5
3	3	9	0	6	7
4	1	4	9	0	8
5	5	7	3	5	0

Bestimmen Sie unter Verwendung des in der Vorlesung angegebenen Algorithmus zur Lösung des TSP-Problems mittels dynamischer Programmierung die Länge der kürzesten Rundreise $g[1, \{2, ..., n\}]$. Geben Sie alle ermittelten Zwischenwerte $g[i, S]$ und die kürzeste Rundreise an.

20 (5 Punkte) Fraktionales Hamsterproblem

Ein Hamster moĉhte die Kapazität seines Maules von $c = 900$[mg] möglichst gut ausnutzen, um für schlechte Zeiten vorzusorgen. Er hat gemäß folgender Tabelle sechs Leckereien zur Auswahl, die er bereits mit indiviudellen Leckerheitsgraden versehen hat. Jede der Leckereien steht genau 1 mal zur Verfügung.
Eine praktische Visualisierung des Problems ist hier verfügbar:
`https://www.youtube.com/watch?v=Ms3QdGIzltU.`

Objekt i	Gewicht t_i [mg]	Leckerheitsgrad p_i
1: Maiskorn	200	15
2: Sonnenblumenkern	50	12
3: Möhrenstück	700	28
4: Getreidekorn	100	8
5: Erdnuss	300	17

Nehmen Sie an, dass der Hamster einen beliebigen Bruchteil $x_i \in [0, 1]$ eines jeden Objekts i aufnehmen kann. Betrachten Sie das Problem somit als fraktionales Rucksackproblem und lösen Sie es mit dem Verfahren aus der Vorlesung.
Geben Sie zu jedem Objekt den Nutzen u_i (als Dezimalzahl mit drei Nachkommastellen) und den gewählten Bruchteil x_i an. Berechnen Sie hieraus den optimalen Gesamtgewinn. Geben Sie zusätzlich die Reihenfolge an, in der der Hamster die Stücke aufsammelt, wenn er sich genau an den Algorithmus hält.

21 (6 Punkte) 0/1-Rucksackproblem

Ein Containerschiff mit einer Kapazität von 900 Tonnen soll mit Containern beladen werden. Jeder Container existiert genau einmal und es können nur vollständige Container auf das Schiff geladen werden. Der Gesamtwert der Ladung soll maximiert werden, ohne dabei die Kapazität zu überschreiten.

Objekt i	Gewicht t_i (Tonnen)	Wert p_i (Perlen)
1: Dreiköpfige Affen	300	17
2: Blaue Pilze	100	14
3: Orange Brechstangen	500	17
4: Grüne Klempneranzüge	100	5
5: Kuchen	800	25

Benutzen Sie Branch&Bound zur Lösung des 0/1-Rucksackproblems analog zum Beispiel aus der Vorlesung. Geben Sie jeweils direkt vor Ausführung von A=POP(S), den aktuellen Wert von Bound b, den Wert der unteren Schranke g(TOP(S)), und den Inhalt des Stacks an.
Mit welcher Beladung kann ein maximaler Transportwert erreicht werden?
Beachten sie: Es gibt Fälle, in denen A nur ein einziges Element x enthält, also $A = \{x\}$, obwohl in der Maske noch Sterne vorkommen; das berücksichtigt der Algorithmus!