

<p>Algorithmen und Berechnungskomplexität I, WS 12/13 Aufgabenblatt 1 Universität Bonn, Institut für Informatik, Abteilung I</p>
--

- Die Lösungen können bis Dienstag, 23.10., 12:15 Uhr in den Postkasten im AVZ III eingeworfen werden (vom Haupteingang im kleinen Raum auf der linken Seite). Gebt bitte immer gut sichtbar auf dem Deckblatt die Gruppennummer (A-I) an, wie auf der Vorlesungswebseite angegeben.
- Abgabe in festen Gruppen von bis zu 3 Personen ist erlaubt.
- Hier kann man sich zu unserer Mailingliste anmelden: <https://lists.iai.uni-bonn.de/mailman/listinfo.cgi/vl-algber1>
- Wer noch keiner Übungsgruppe zugeordnet ist und dennoch am Übungsbetrieb teilnehmen möchte, kontaktiert bitte Rainer Penninger (penninge@cs.uni-bonn.de).

Aufgabe 1: Rekursionsgleichungen (4 Punkte)

Sei eine rekursive Kostenfunktion T definiert durch

$$T(n) = T(n - 1) + 2 \cdot n + 3$$

und $T(0) = 0$. Beweisen Sie mittels vollständiger Induktion über n , dass $T(n) \in \Theta(n^2)$.

Aufgabe 2: O-Notation (4 Punkte)

Ist die Aussage

$$\frac{n^{15}}{3^n} \in O(1)$$

korrekt? Begründen Sie Ihre Antworten.

Bitte wenden!

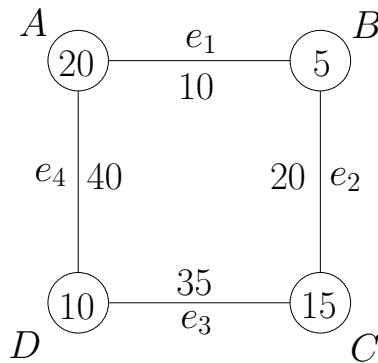


Abbildung 1: Der zu traversierende Graph G mit Startknoten A .

Aufgabe 3: Graphtraversierung (4 Punkte)

Gegeben sei der Graph G in Abbildung 1. Wie in der Vorlesung beschrieben sollen auf allen Knoten Agenten platziert werden. Alle Agenten starten auf Knoten A . Zusätzlich darf angenommen werden, dass die nicht platzierten Agenten immer auf *einem* Knoten stehen, d.h. sich nicht in mehrere Gruppen aufteilen.

Geben Sie eine optimale Traversierungsstrategie an. Das heißt konstruieren Sie eine Strategie, die G mit einer gewissen Anzahl Agenten traversiert. Zeigen Sie weiterhin, dass G mit weniger Agenten nicht traversiert werden kann.

Tipp: Hierbei kann man entweder alle Strategien auflisten und die Beste auswählen, oder mit etwas Knobeln eine gute Strategie erraten und zeigen, dass mit weniger Agenten G nicht traversiert werden kann.

Aufgabe 4: Insertionsort (4 Punkte)

Sortieren Sie die folgende Zahlenfolge mit Hilfe des Insertion-Sort-Algorithmus:

56, 34, 47, 2, 55, 87, 74, 28, 104, 6, 3, 65, 51, 62, 58, 12

Geben Sie dabei in Form einer Tabelle für jeden Schritt das sortierte Teilergebnis an sowie die Anzahl der Vertauschungen, die dazu nötig waren.