

1. Aufgabe: Graphen

5+5+10 Punkte

- a) Sei $G = (V, E)$ ein Graph und $\delta(v)$ der Grad von Knoten $v \in V$.
Zeige: $\sum_{v \in V} \delta(v) = 2|E|$.
- b) Hat Graph H aus Abbildung 1 einen Hamiltonpfad? (mit Begründung)

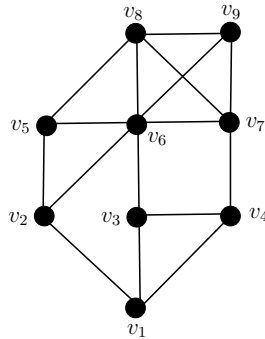


Abbildung 1: Der Graph H .

- c) Wende Breitensuche auf den Graphen H aus Abbildung 1 an; starte dabei mit dem Knoten v_1 . Falls zu einem Zeitpunkt mehrere Knoten für den nächsten Schritt in Frage kommen, wähle denjenigen mit dem kleinsten Index. Gib die Menge Q jedesmal wenn sie sich ändert an und zeichne den gefundenen Baum T .

2. Aufgabe: Datenstrukturen

8+6+6 Punkte

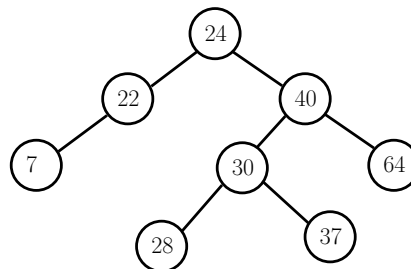


Abbildung 2: Der AVL-Baum T .

- a) Gegeben sei der AVL-Baum T aus Abbildung 2. Füge nacheinander die Elemente 23 und 26 ein, so dass T ein AVL-Baum bleibt; gib den Baum nach jeder Einfüge-Operation an.
(Hinweis: Nach *jedem* Einfügen soll der geänderte Baum ein AVL-Baum sein. Zum Schluss sollen beide Zahlen eingefügt sein.)

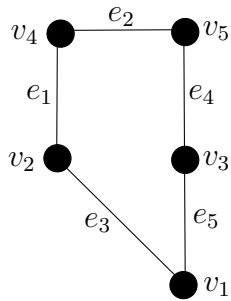


Abbildung 3: Der Graph G .

- b) Stelle die Adjazenzmatrix und die Inzidenzmatrix zum Graphen G aus Abbildung 3 auf.
- c) Ergänze die fehlenden Anweisungen in LIST-DELETE für doppelt-verkettete Listen:

```
LIST-DELETE(L, x)
  if vorg[x] ≠ NIL
    then ...
    else ...
  if nachf[x] ≠ NIL
    then ...
```

3.Aufgabe: Komplexität

8+7 Punkte

- a) Seien $f, g : \mathbb{N} \mapsto \mathbb{R}$ zwei Funktionen. Zeige:

$$f \in \Theta(g) \Leftrightarrow f \in O(g) \text{ und } f \in \Omega(g).$$

- b) Zeige: $8n^6 + 11 \in O(n^9)$. Gib dazu explizit geeignete Konstanten c und n_0 aus der Definition an und zeige, dass sie die Definition erfüllen.

4.Aufgabe: Rekursionen

8+7 Punkte

- a) Finde mit Hilfe von erzeugenden Funktionen einen geschlossenen Ausdruck für die Rekursion $a_n = 2a_{n-1}$, $n = 1, 2, \dots$ und $a_0 = 1$.
- b) Bestimme mit Hilfe des Mastertheorems das asymptotische Wachstum der Rekursion

$$T(n) = 27 \cdot T\left(\frac{n}{3}\right) + n^3.$$

5.Aufgabe: Hashing

10 Punkte

Wir betrachten ein leeres Array A der Größe 9, d.h. es gibt die Speicherzellen $A[0], A[1], \dots, A[8]$; in diesem führen wir offenes Hashing mit linearem Sondieren mit der folgenden Sondierungsfunktion durch:

$$t(i, x) = (x + i) \bmod 9$$

(also eine vereinfachte Kurzform von $t(i, h(x)) = (h(x) + i) \bmod 9$ mit $h(x) = x \bmod 9$). Dabei ist x ein einzusetzender Schlüssel und i die Nummer des Versuches, x in eine unbesetzte Speicherzelle des Arrays zu schreiben.

Gib zu jedem der folgenden Schlüssel die Position an, die er in A bekommt:

15, 18, 42, 6, 51

Gib außerdem das Array nach dem Einfügen aller Schlüssel an.

(Hinweis: Die Schlüssel sollen in der gegebenen Reihenfolge eingefügt werden.)

6.Aufgabe: Sortieren

10 Punkte

Gegeben sei das folgende Array A :

$$A[1] = 8 \quad A[2] = 7 \quad A[3] = 3 \quad A[4] = 6$$

Wende Mergesort auf das Array A an.

Gib die Aufrufe der Funktionen *Mergesort* und *Merge* in zeitlicher Abfolge mit den jeweiligen Parametern an. Gib außerdem das Array am Ende jedes Aufrufes von *Merge* an.

7.Aufgabe: Suche in Graphen

2+2+2+2+2 Punkte

- a) Eine Warteschlange arbeitet nach dem FIFO-Prinzip. wahr
 falsch
- b) Eine Permutation der Zahlen $1, 2, \dots, n$ kann in $O(n)$ sortiert werden. wahr
 falsch
- c) Verwendet man für die Suche nach einer Zahl x in einer Menge von n sortierten Zahlen die *Binäre Suche*, so kann dieses im Worst Case $O(n^2)$ dauern. wahr
 falsch
- d) Die Breitensuche hat die gleiche Laufzeit wie die Tiefensuche (im Worst Case, in asymptotischer Notation). wahr
 falsch
- e) Ein Eulerweg besucht jeden Knoten in einem Graphen genau einmal. wahr
 falsch

Viel Erfolg!!!