

Klausur: Informatik II

21.6.2002

1. Gegeben sei die folgende Adjazenzmatrix eines gerichteten Graphen:

| | A | B | C | D | E | F | G | H |
|---|---|---|---|---|---|---|---|---|
| A | | 3 | 1 | 2 | | | | |
| B | | | | | | | | |
| C | | | | | 1 | | | |
| D | | | | | 1 | | | |
| E | | | | | | 1 | 2 | |
| F | | | | | | | | 3 |
| G | | | | | | | | 1 |
| H | | | | | | | | |

Bestimmen Sie mit einer Variante Ihrer Wahl zum Algorithmus von Dijkstra den jeweils kürzesten Weg von A zu allen übrigen Knoten.

2. Entscheiden Sie sich zunächst, ob Sie für die folgende Aufgabe Quick Sort oder Heap Sort verwenden wollen.

Zeigen Sie, wie der Algorithmus zum Sortieren der Folge

T I P P E R A R Y

verwendet wird.

3. Skizzieren Sie das Einfügen der Schlüssel

4, 16, 14, 7, 3, 11, 12, 19, 13

- in einen freien knotenorientierten binären Suchbaum
- in einen AVL-Baum
- in eine Hash Tabelle mit Hash Funktion $\text{hash}(x) = x \bmod 11$ und open hashing als Kollisionsauflösungsstrategie.

Berechnen Sie die mittlere Zeitkomplexität (Anzahl Vergleiche, Einfach- und Doppelrotationen und Berechnungen der Hash Funktion).

4. Diskutieren Sie die Vor- und Nachteile von Hashing im Vergleich zu Suchverfahren, die auf der Verwendung von Suchbäumen beruhen.

5. Welche Implementierung (dynamischer Vector oder Liste) würden Sie für folgende Abstrakte Datentypen vorziehen?

- Stack
- Queue
- Priority Queue

Bitte geben Sie eine kurze Begründung!

6. Ein binärer Suchbaum liefert beim Durchlaufen in Schichtenordnung (erst die Wurzel, dann alle Knoten der Höhe 1 von links nach rechts, dann alle Knoten der Höhe 2 von links nach rechts, ...) die Ausgabe

C A F B E H G.

Wie sieht der Baum aus?

Welcher Algorithmus zum Durchlaufen von Graphen entspricht dem Durchlaufen von Bäumen in Schichtenordnung?