

Aufgabe 1 [1.5]

- a. [0.9] Erstellen Sie in der Programmiersprache C++ Programme oder Programmteile die folgendes Laufzeitverhalten zeigen:
1. $O(n^3)$
 2. $O(\log n)$
 3. $O(n^3 * \log(n))$
- b. [0.6] Was versteht man bei der Laufzeitabschätzung eines Programms unter $O(n)$, $\Omega(n)$ und $\Theta(n)$?

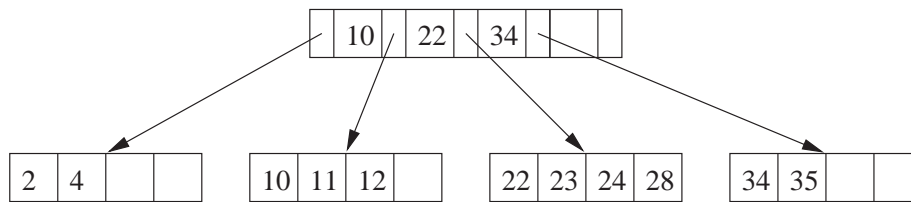
Aufgabe 2 [0.9]

Erläutern Sie die Algorithmenentwurfsverfahren *greedy*, *divide-and-conquer* und *dynamic programming* anhand ihrer

- zugrundeliegenden Ideen
- ihrer Vor- und Nachteile und
- anhand eines kurzen Beispiels.

Aufgabe 3 [2.4]

Gegeben ist der folgende B^+ -Baum der Ordnung 2.



Fügen Sie die Werte 9, 14, 29 und 15 (in dieser Reihenfolge) ein und löschen Sie danach die Werte 10 and 15 (in dieser Reihenfolge).

Geben Sie alle Zwischenschritte bei der Veränderung des Baumes an.

Aufgabe 4 [2.0]

Gegeben ist ein Heap zur Verwaltung von Minima durch die folgende Feldspeicherung (d.h. Kinder an Position $2i+1$ und $2i+2$):

| 1 | 4 | 3 | 7 | 6 | 5 | = Heap-Elemente

| 0 | 1 | 2 | 3 | 4 | 5 | = Feld-Position

- [0.1] Stellen Sie die Baumstruktur des gegebenen Heaps graphisch dar.
- [0.6] Geben Sie 2 Eingabereihenfolgen der im obigen Heap gespeicherten Werte an, die die Struktur des gegebenen Heaps erzeugen.
- [1.0] Tragen Sie die Elemente 0, 2, 8 (in dieser Reihenfolge) in den Heap ein und löschen Sie danach 2x das Minimum. Geben Sie alle signifikanten Zustände während des Einfügens und Löschens anhand der Baumdarstellung an.
- [0.3] Geben Sie das Laufzeitverhaltens der Heap-Operationen Einfügen, Löschen und Zugriff auf Minimum an.

Aufgabe 5 [1.2]

Gegeben ist folgender Zahlenvektor:

567,123, 432, 234, 321, 765, 999, 111, 012, 978

Sortieren Sie diese Zahlfolge mit Hilfe des

- Binären Quicksorts, und des
- LSD Radix Sort.

Geben Sie alle Zwischenstufen des Zahlenvektors während der Sortiervorgänge an.

Aufgabe 6 [1.0]

Gegeben ist ein ungerichteter Graph $G=(V,E)$, wobei $V = \{1, 2, 3, 4, 5, 6\}$ und $E = \{[1, 2], [1, 3], [1, 4], [1, 5], [2, 3], [3, 4], [4, 5]\}$ ist.

- a. [0.1] Stellen Sie diesen Graphen zeichnerisch dar.
- b. [0.1] Speichern Sie diesen Graphen mit einer Adjazenzmatrix und geben Sie diese an.
- c. [0.8] Traversieren Sie diesen Graphen mit dem dfs (depth-first-search) und dem bfs (breadth-first-search) Verfahren. Geben Sie die Reihenfolge der besuchten Knoten an und erläutern Sie Ihre Vorgangsweise durch Verwaltung der zu besuchenden Knoten mit einer entsprechenden Datenstruktur (entsprechend der Vorlesung).

Aufgabe 7 [1.0]

m Missionare und k Kannibalen befinden sich am Ufer eines Flusses im Regenwald. Ihnen steht ein Boot zur Verfügung, das b Personen Platz bietet. Es ist ein Verfahren zu entwickeln, das es erlaubt, alle $m+k$ Personen überzusetzen, wobei die Nebenbedingung einzuhalten ist, daß zu keiner Zeit an keinem Ort die Kannibalen die Mehrheit erhalten dürfen, da diese sonst im Rahmen einer sozio-kulturellen Regression die Missionare verspeisen würden.

- a. [0.5] Modellieren Sie zunächst die Situation als Zustandsraum. Identifizieren Sie Start- und Zielzustände sowie erlaubte Zustandsübergänge. Geben Sie für ein konkretes Beispiel $\langle b, k, m \rangle$ die ersten 5 Konfigurationen des Zustandsraumes als gerichteten Graph an.
- b. [0.5] Skizzieren Sie in Pseudosprache die Suche nach dem kürzesten Weg im Zustandsraum aus a), das heißt, es soll jene (erlaubte!) Übersetzungsfolge gefunden werden, die mit der geringsten Anzahl an Flußüberquerungen auskommt.