Prof. Dr. Klaus Volbert

Regensburg, den 04. November 2025
Votieren und abgeben bis zum 10. November 2025, 23:00 Uhr

Übungen zur Vorlesung
Algorithmen und Datenstrukturen
WiSe 2025/26
Blatt 5

Wichtige Hinweise:

−> Falls Sie bei der Bearbeitung einer Aufgabe größere Schwierigkeiten hatten und deswegen die Bearbeitung abgebrochen haben, so versuchen Sie bitte Ihre Schwierigkeiten in Form von Fragen festzuhalten. Bringen Sie Ihre Fragen einfach zur Vorlesung oder zur Übung mit!

−> Musterlösungen werden bei Bedarf in den Übungen besprochen!

Aufgabe 1:
Demonstrieren Sie die Funktionsweise von MergeSort und HeapSort anhand des Feldes $a[] = \{-5, 13, -32, 7, -3, 17, 23, 12, -35, 19\}$. Führen Sie die Demonstration von MergeSort anhand eines Rekursionsbaums durch, indem Sie den Baum in der Phase „Teilen“ von oben nach unten aufbauen und anschließend in der Phase „Mischen“ von unten nach oben durchlaufen. Nennen Sie jeweils die entstandenen Teilfolgen. Führen Sie bei der Demonstration von HeapSort sowohl entstehende Bäume als auch entstehende Felder mit. Überprüfen Sie Ihre Ausführungen mit Hilfe eines C, C++, C#, Java oder Python-Programms, das die beiden Sortierverfahren implementiert und die wesentlichen Informationen ausgibt.

Aufgabe 2:
Zeigen Sie folgende Aussagen:

1. Ein Heap mit n Elementen hat die Höhe $\lfloor \log n \rfloor$
2. Ein Heap mit n Elementen hat höchstens $\lceil \frac{n}{2^{h+1}} \rceil$ viele Knoten der Höhe h
3. Für alle x mit $|x| < 1$: $\sum_{k=0}^{\infty} kx^k = \frac{x}{(1-x)^2}$

   (Tipp: Differenzieren Sie beide Seiten der geometrischen Reihe und multiplizieren Sie mit x)

Kann die Reihenfolge der Heapify-Aufrufe in der Operation BuildHeap ohne weitere Modifikation vertauscht werden? Begründen Sie Ihre Antwort!

Aufgabe 3:
Stellen Sie sich vor, Sie sollen zwei quadratische Matrizen $M, N \in \mathbb{R}^{n \times n}$ miteinander multiplizieren. Sei $n = 2^i$ für ein $i \in \mathbb{N}$ die Dimension der Matrizen, dann kann man M, N und $O = M \cdot N$ wie folgt zerlegen mit $M_{ij}, N_{ij}, O_{ij} \in \mathbb{R}^{n/2 \times n/2}$ für $i, j \in \{1, 2\}$:

$$M := \begin{pmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{pmatrix}, N := \begin{pmatrix} N_{11} & N_{12} \\ N_{21} & N_{22} \end{pmatrix}, O := \begin{pmatrix} O_{11} & O_{12} \\ O_{21} & O_{22} \end{pmatrix}$$

1. Zeigen Sie, dass die beiden folgenden Varianten die Produktmatrix O korrekt berechnen:

- Variante 1:

$$\begin{aligned}
O_{11} &:= M_{11} \cdot N_{11} + M_{12} \cdot N_{21} \\
O_{12} &:= M_{11} \cdot N_{12} + M_{12} \cdot N_{22} \\
O_{21} &:= M_{21} \cdot N_{11} + M_{22} \cdot N_{21} \\
O_{22} &:= M_{21} \cdot N_{12} + M_{22} \cdot N_{22}
\end{aligned}$$

- Variante 2:

$$\begin{aligned}
H_1 &:= (M_{11} + M_{22}) \cdot (N_{11} + N_{22}) \\
H_2 &:= (M_{21} + M_{22}) \cdot N_{11} \\
H_3 &:= M_{11} \cdot (N_{12} - N_{22}) \\
H_4 &:= M_{22} \cdot (N_{21} - N_{11}) \\
H_5 &:= (M_{11} + M_{12}) \cdot N_{22} \\
H_6 &:= (M_{21} - M_{11}) \cdot (N_{11} + N_{12}) \\
H_7 &:= (M_{12} - M_{22}) \cdot (N_{21} + N_{22})
\end{aligned}$$

$$\begin{aligned}
O_{11} &:= H_1 + H_4 - H_5 + H_7 \\
O_{12} &:= H_3 + H_5 \\
O_{21} &:= H_2 + H_4 \\
O_{22} &:= H_1 - H_2 + H_3 + H_6
\end{aligned}$$

2. Bestimmen Sie die asymptotische Laufzeitkomplexität beider Varianten und vergleichen Sie diese mit der Komplexität der Standardmethode zur Multiplikation zweier Matrizen.

Aufgabe 4:
Implementieren Sie die Variante 2 aus Aufgabe 3 in C, C++, C#, Java oder Python. Ist Ihre Implementierung schneller als die Standard-Methode? Wenn ja, ab welcher Dimension n?

2.1) Sei H ein Heap mit Höhe h und n Elementen:

$$2^h \le n \le 2^{h+1} - 1 < 2^{h+1}$$

$$\Rightarrow h \le \log n < h+1$$

$$\Rightarrow h = \lfloor \log(n) \rfloor$$

3) $$\left(\sum_{k=0}^{\infty} x^k \right)' = \left(\frac{1}{1-x} \right)' \quad \text{für } x+1$$

$$x \cdot \left(\sum_{k=0}^{\infty} k x^k \right) = \frac{x}{(1-x)^2}$$

2.) Beh.: In H gibt es max $\left\lceil \frac{n}{2^{h+1}} \right\rceil$ viele Knoten der Höhe H

IA: $h = 0$

Blätter in H: $n - \left(\left\lfloor \frac{(n-1)-1}{2} \right\rfloor \right) = n - \left(\left\lfloor \frac{n-2}{2} \right\rfloor + 1 \right)$

$$= \begin{cases} n - \left(\frac{n-2}{2} + 1 \right) = \frac{n}{2} = \left\lceil \frac{n}{2} \right\rceil & \text{1. Fall } n \text{ gerade} \\ n - \left(\frac{n-3}{2} + 1 \right) = \frac{n}{2} + \frac{1}{2} = \frac{\lceil n \rceil}{2} & \text{2. Fall sonst} \end{cases}$$

IV: Beh gilt für $h-1$

IS: $h-1 \Rightarrow h$

Aus dem Heap H entsteht durch Streichen aller Blätter der Heap H'. Beobachtung: Jeder Knoten in H' mit Höhe $h-1$ hat in H die Höhe h

$$\frac{\left\lceil n - \left\lceil \frac{n}{2} \right\rceil \right\rceil}{2^{(h-1)+1}} = \frac{\left\lceil \left\lfloor \frac{n}{2} \right\rfloor \right\rceil}{2^{h}} \leq \frac{\left\lceil \frac{n}{2} \right\rceil}{2^{h}} = \frac{\lceil n \rceil}{2^{h+1}}$$

1) (-5, 13, -32, 7, -3, 17, 23, 12, -35, 19)

(-5, 13, -32, 7, -3) (17, 23, 12, -35, 19)

(-5, 13, -32) (7, -3) (17, 23, 12) (-33, 19)

(-5, 13) (-32) (7) (-3) (17, 23) (12) (-33) (19)

(-5) (13) (-32) (7) (-3) (17) (23) (12) (-33) (19)

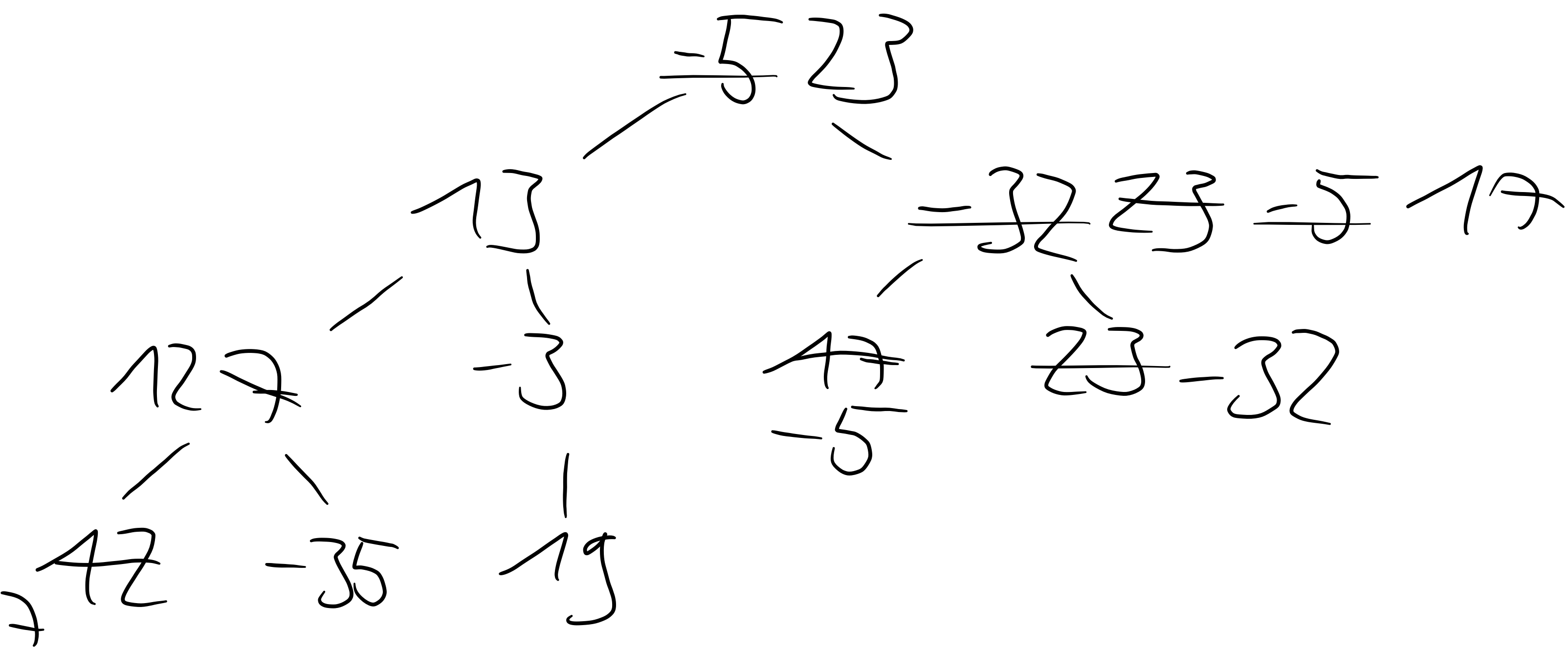
(-5, 13) (-32, 7) (-3, 17) (12, 23) (-33, 19)

(-32, -5, 7, 13) (-3, 12, 17, 23) (-33, 19)

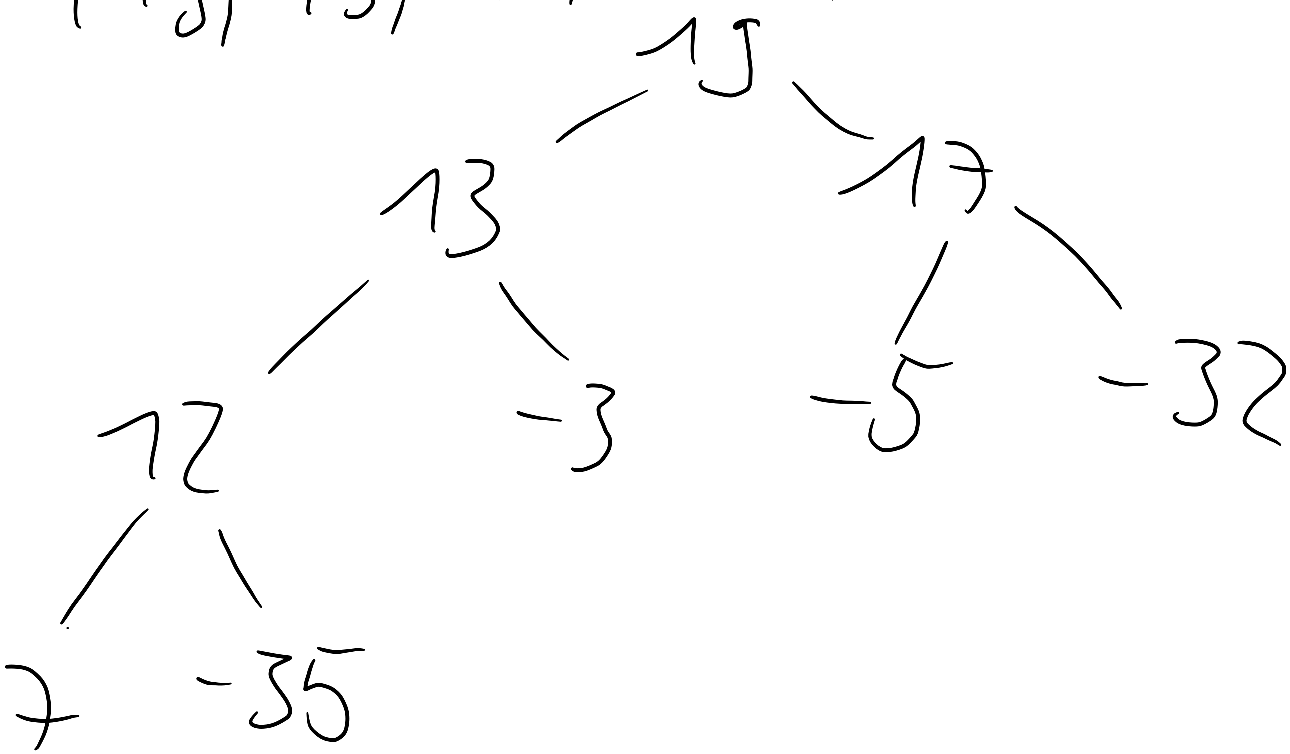
(-32, -5, -3, 7, 12, 13, 17, 23) (-33, 19)

(-33, -32, -5, -3, 7, 12, 13, 17, 19, 23)

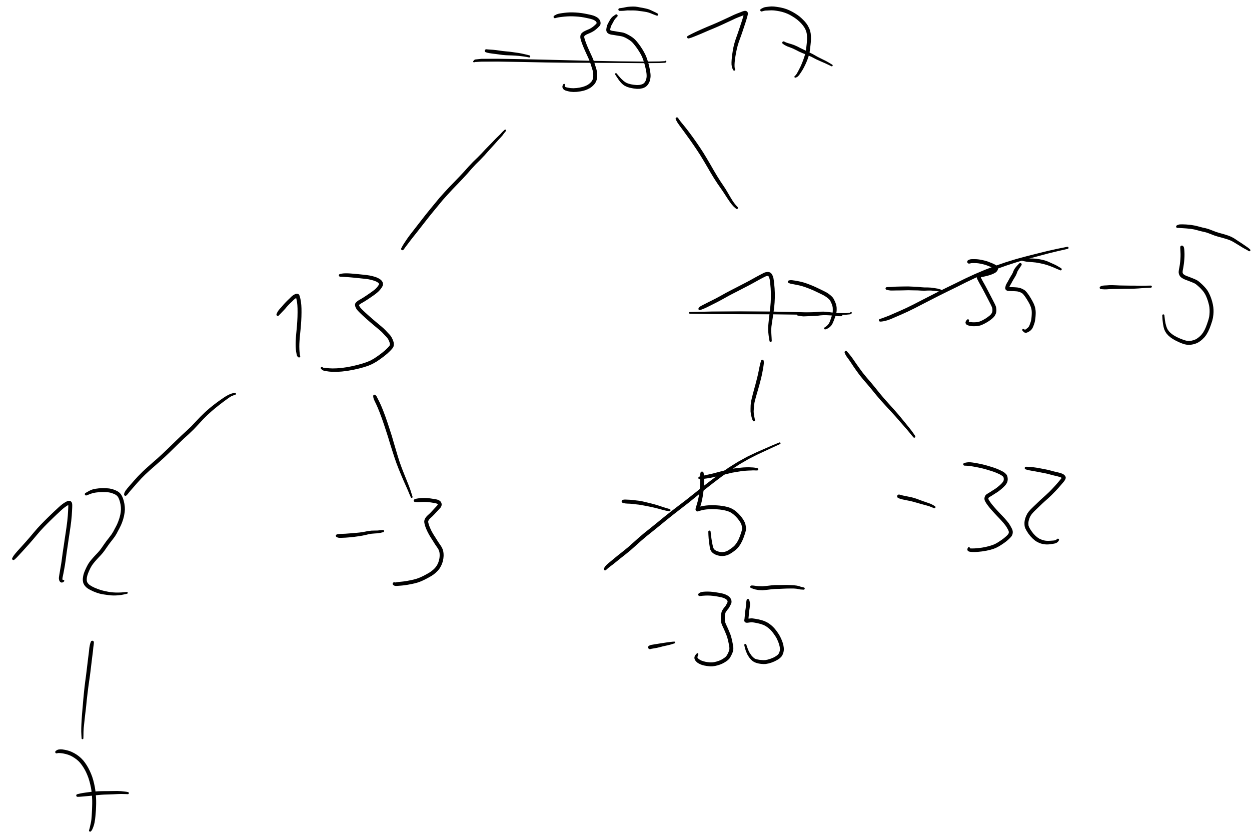
1.2)



(19, 13, 17, 2, -3, -32, 7, -39, 23)



(-35, 13, 17, 12, -3, -5, -32, 7, 19, 23)



(7, 13, -5, 12, -3, -35, -32, 17, 19, 23

...