

Aufgabe 32

- (b) Gesucht ist ein Algorithmus, der die $\lceil \log n \rceil$ bzw. $\lceil \sqrt{n} \rceil$ Kunden mit den meisten Flügen bestimmt.

Unser Algorithmus sieht wie folgt aus:

1. Erstelle aus der Kundendatenbank einen Heap, genauer gesagt einen maxHeap.
2. Sortiere den Heap wie bekannt, allerdings nur, bis die ersten $\lceil \sqrt{n} \rceil$ Kunden mit den meisten Flügen rausgefiltert sind.

Da $\log n \in \mathcal{O}(\sqrt{n})$ reicht es aus, die ersten $\lceil \sqrt{n} \rceil$ Kunden zu bestimmen. So erhält man die Liste der Kunden mit den meisten Flügen.

Die Laufzeit dieses Algorithmus' sieht wie folgt aus: Das erstellen des Heaps benötigt $\mathcal{O}(n)$ Laufzeit. Das "rausnehmen" der Elemente erfordert $\sqrt{n} \cdot \log n$ Laufzeit. Also gilt insgesamt:

$$T(n) = f + \underbrace{\sqrt{n} \cdot \log n}_{\in \mathcal{O}(\sqrt{n} \cdot \sqrt{n}) = \mathcal{O}(n)} \in \mathcal{O}(n) \quad \text{mit } f \in \mathcal{O}(n)$$

(Die Gaußklammer $\lceil \cdot \rceil$ kann weggelassen werden, da sie in der asymptotischen Betrachtung keine Rolle spielt.)

Dementsprechend erfüllt dieser Algorithmus die Anforderungen.

- (a) Da $\mathcal{O}(n) \subset \mathcal{O}(n \cdot \log n)$ erfüllt der Algorithmus in (b) auch diese Anforderungen¹.

¹Falls wir uns mit unserem Algorithmus in (b) vertan haben sollten, würden wir ganz einfach Heap-Sort für (a) verwenden, welches bekannterweise in $\mathcal{O}(n \cdot \log n)$ liegt ;)