

**Präsenzaufgabe 6.1**

Gegeben sei ein leerer Heap.

- a) Fügen Sie die Elemente 17, 23, 2, 11, 48, 45 und 8 in dieser Reihenfolge ein und stellen Sie diesen nach jedem Schritt grafisch dar. Stellen Sie den entstandenen Heap am Ende auch in einem Array dar.
- b) Entfernen Sie aus dem resultierenden Heap zweimal das Minimum und stellen Sie den sich ergebenden Heap jeweils dar.

**Präsenzaufgabe 6.2** (4 Punkte)

Gegeben sei die folgende alternative Prozedur zum Erstellen eines binären Heaps für ein unsortiertes Array  $A[1..n]$ :

*buildHeapInsert*( $A : \text{Array}$ )

```
1  for  $i := 1$  to  $n$   
2      do insert( $A[i]$ )
```

- a) Zeigen Sie, dass *buildHeapInsert* im schlechtesten Fall die Laufzeit  $\Theta(n \log n)$  benötigt, um einen Heap mit  $n$  Elementen zu erzeugen.
- b) Erzeugen die Prozedur *buildHeapBackwards* aus der Vorlesung und *buildHeapInsert* immer den gleichen Heap, wenn sie auf das gleiche Eingabefeld angewendet werden? Beweisen Sie, dass sie dies tun, oder geben Sie ein Gegenbeispiel an.