

---

## Algorithmen und Berechnungskomplexität I WS 15/16

Universität Bonn, Institut für Informatik, Abteilung I

7. Aufgabenblatt zur Vorlesung

Abgabe: 08.12. (12<sup>30</sup>)

---

### Aufgabe 25: Einen Algorithmus analysieren (4 Punkte)

Betrachten Sie den folgenden rekursiven Algorithmus, der auf einem Array mit natürlichen Zahlen operiert:

**Algorithmus** *QS(array A)*  
QSrek(A, 1, size(A));

**Algorithmus** *QSrek(Array A, integer l, r)*

if  $l < r$  then

$p := \text{part}(A, l, r)$ ;

    QSrek(A, l, p-1);

    QSrek(A, p+1, r);

**Algorithmus** *part(A, l, r)*

$x := A[r]$ ;

$i := l - 1$ ;

for  $j = l$  to  $r - 1$  do

    if  $A[j] \leq x$  then

$i := i + 1$ ;

        Vertausche  $A[i]$  und  $A[j]$ ;

Vertausche  $A[i + 1]$  mit  $A[r]$

return  $i + 1$ ;

Vollziehen Sie die Funktionsweise des Algorithmus anhand eines Beispielarrrays mit mindestens 10 Einträgen nach. Was macht dieser Algorithmus?

### Aufgabe 26: Suchbäume (4 Punkte)

Zeigen Sie, dass für einen Binärbaum mit paarweise disjunkten Schlüsseln gilt: Die Suchbaumeigenschaft gilt genau dann, wenn der In-Order-Durchlauf die Schlüssel in aufsteigend sortierter Reihenfolge ausgibt.

**Aufgabe 27: Aufbau eines AVL-Baums (4 Punkte)**

Geben Sie den AVL-Baum an, der durch sukzessives Einfügen der Schlüssel-  
folge

4, 5, 7, 2, 1, 3, 6

in einen anfangs leeren Baum entsteht.

Dokumentieren Sie dabei auch die Zwischenergebnisse und die durchgeführ-  
ten Umstrukturierungen.

**Aufgabe 28: AVL-Baum: Elemente löschen (4 Punkte)**

Löschen Sie aus dem abgebildeten *AVL*-Baum den Knoten mit Schlüssel *C*.  
Beschreiben Sie wie der Knoten entfernt wird und welche Schritte anschlie-  
ßend zum Rebalancieren durchgeführt werden.

