

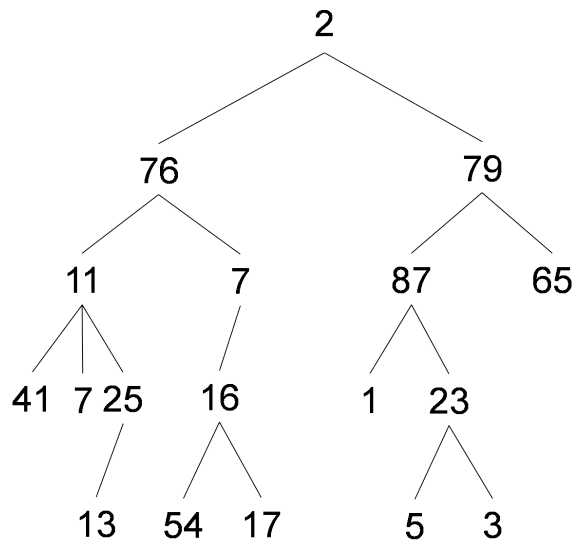
Übungsserie Nr. 2

Ausgabe: 04. März 2009
 Abgabe: 11./12. März 2009

1. Aufgabe: (8 Punkte) Wurzelbäume

In der Vorlesung haben Sie die Klammerdarstellung sowie die Darstellung in eingerückter Form und als (gerichteten) Graphen eines Wurzelbaums kennengelernt.

(1a) (2 Punkte) Geben Sie zu dem folgenden Baum die zugehörige Klammerdarstellung und Darstellung in eingerückter Form an:



(1b) (2 Punkte) Geben Sie zur folgenden Klammerdarstellung den zugehörigen Baum als gerichteten Graphen und in eingerückter Form an:

$S(R(H(K)), P(A(N, O), Q, T), V(J, F(G)))$

(1c) (1 Punkt) Können Sie aus den obigen Klammerdarstellungen den dazugehörigen Baum eindeutig rekonstruieren? Begründen Sie Ihre Antwort.

(1d) (3 Punkte) Beantworten Sie für die Bäume aus Teilaufgaben a) und b) folgende Fragen:

- i) Was ist die Höhe des Baumes? (Ein Baum mit nur *einem* Knoten habe die Höhe 1.)
- ii) Welches sind die längsten Pfade in diesem Baum?
- iii) Welche Knoten bilden die Blätter dieses Baumes?

2. Aufgabe: (8 Punkte) Sortieren mit Arrays

Schreiben und testen Sie ein Java-Programm, welches ein Array von Zahlen sortiert. Nehmen Sie das Gerüst `SortArray.java` von der Vorlesungs-Webseite als Vorlage und gehen Sie dabei folgendermassen vor:

(2a) (2 Punkte) Schreiben Sie eine `initialize`-Methode, die das Array mit 10 Zufallszahlen vom Typ `int` initialisiert. Benutzen Sie dazu die Klasse `java.util.Random`, die Sie wie folgt verwenden können:

```
import java.util.Random; // Importiert "Random" Klasse
...
Random r = new Random(); // erstellt Instanz r
                        // des Zufallsgenerators
...
int i = r.nextInt(1000); // gibt Zufallszahl zurueck
```

Sie können so gleichverteilte Zufallszahlen zwischen 0 und 1000 generieren. Die API-Dokumentation des Pakets `java.util` finden Sie auf der Sun-Webseite (<http://java.sun.com/j2se/1.5.0/docs/api/>).

(2b) (2 Punkte) Schreiben Sie zu Testzwecken eine `print`-Methode, die den Inhalt des `int`-Arrays ausgibt.

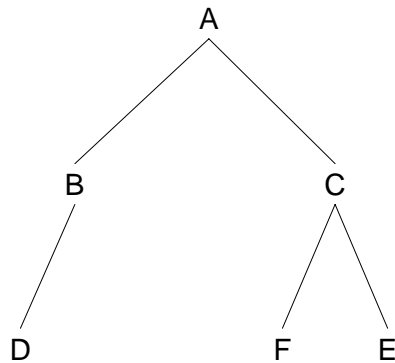
(2c) (4 Punkte) Schreiben Sie eine rekursive Methode `sort`, die das Array in absteigender Reihenfolge **rekursiv** sortiert. Die Methode soll einen Parameter `i` haben, der den Beginn des zu sortierenden Bereiches markiert. Nehmen Sie an, dass alle Elemente des Arrays mit Index $< i$ bereits sortiert sind. Suchen Sie nun das grösste Element des übrigen Arrays (mit Index $j > i$) und vertauschen Sie es mit dem Element mit Index i . Rufen Sie anschliessend Ihre Sortierfunktion rekursiv auf, um den Rest des Arrays zu sortieren. Nach erfolgter Sortierung soll für die Elemente des Arrays also $a[0] \geq a[1] \geq a[2] \geq \dots \geq a[9]$ gelten. Überlegen Sie sich:

- Wie die Abbruchbedingung für die Rekursion lauten muss.
- Wie sich der Parameter `i` bei jedem rekursiven Aufruf ändern muss.
- Wie die Sortiermethode vom Hauptprogramm (`main`) aufgerufen werden muss.

3. Aufgabe: (6 Punkte) Binärbaum als Array

Ein Binärbaum ist ein Wurzelbaum, in dem jeder Knoten höchstens zwei Nachfolger hat. Binärbäume kann man leicht in einem Array speichern, wenn dieses geeignet interpretiert wird. Die Idee besteht darin, die Wurzel an Stelle 1 des Arrays zu setzen und die beiden direkten Nachfolger von Stelle i an den Stellen $2i$ und $2i + 1$ zu speichern.

Es lässt sich dann beispielsweise ein Baum folgendermassen “niveauweise” speichern:



```
char[] tree = new char[n];  
  
tree[1] = 'A';  
tree[2] = 'B';  
tree[3] = 'C';  
tree[4] = 'D';  
tree[5] = ' ';  
tree[6] = 'F';  
tree[7] = 'E'; ...
```

Hier wird für einen “leeren Unterbaum” ein Leerzeichen eingetragen.

(3a) (6 Punkte) Schreiben Sie nun ein Java-Programm, das einen als char-Array gespeicherten Binärbaum in eingerückter Form ausgibt. Vervollständigen Sie dazu die rekursive Methode `indent` der Klasse `ConvertTree` von der Vorlesungs-Webseite. Die Methode soll einen Knoten samt seiner Unterbäume eingerückt auf dem Bildschirm ausgeben. Die Methode besitzt zwei `int`-Parameter: `i` und `j`. Der erste bezeichnet den auszugebenden Knoten durch seine Position im Array, der zweite gibt die Einrücktiefe des bezeichneten Knotens an. Das Eingabe-Array kann dem Programm auf der Kommandozeile als (mit Anführungszeichen umschlossener) String übergeben werden. Hier ein Beispielablauf mit dem obigen Baum:

```
C:\ java ConvertTree "ABCD FE"  
A  
  B  
    D  
  C  
    F  
    E
```

Überlegen Sie sich:

- In welcher Reihenfolge Sie die Knoten eines Baumes ablaufen, um ihn “von Hand” einzurücken.

- Welchen Index im Array der “Vorgänger” (der “linke Nachfolger”, der “rechte Nachfolger”) eines Knotens mit Index i hat.
- Wie Sie einen leeren Baum (bzw. einen leeren Unterbaum) behandeln.
- Wie Sie die Abbruchbedingung formulieren können.

Summe: 22 Punkte