

## Übungsblatt 11

Abgabe bis Freitag, 17.07.2015, 10:00 Uhr

### Hinweis:

Aufgaben immer per E-Mail (eine E-Mail pro Blatt und Gruppe) an den zuständigen Tutor schicken (Bei Programmieraufgaben Java Quellcode und evtl. benötigte Datendateien).

### Aufgabe 11.1

Das pascalsche Dreieck ist eine Form der grafischen Darstellung der Binomialkoeffizienten. Sie sind im Dreieck derart angeordnet, dass jeder Eintrag die Summe der zwei darüberstehenden Einträge ist:

$n = 0:$					1				
$n = 1:$				1		1			
$n = 2:$			1		2		1		
$n = 3:$		1		3		3		1	
$n = 4:$	1		4		6		4		1

Schreiben Sie eine rekursive Java-Methode, die eine Zeile des pascalschen Dreiecks berechnet. Eine Zeile des Dreiecks soll hierbei als Array von int Werten zurückgegeben werden.

```
public class Pascal {  
    Pascal () {}  
    public int[] pascal(int zeile) {  
        \\....  
    }  
}
```

### Aufgabe 11.2

Schreiben Sie eine Klasse `Sort`, die Methoden zum Sortieren von Arrays bereitstellt.

1. Implementieren Sie eine Methode

```
static int[] union(int[] a1, int[] a2),
```

der zwei **sortierte** Folgen von Integer-Zahlen übergeben werden. Diese Methode soll eine sortierte Folge zurückgeben, die alle Elemente aus `a1` und `a2` enthält.

2. Bestimmen Sie die Komplexität der Methode `union` in Abhängigkeit der Länge der Arrays `a1` und `a2`.
3. Implementieren Sie eine **rekursive** Methode  
`static int[] sort(int[] array),`  
die eine Folge von Integer-Zahlen in sortierter Reihenfolge zurückgibt. Gehen Sie dabei wie folgt vor:
  - (a) Unterteilen Sie die Folge in zwei möglichst gleich große Teilfolgen.
  - (b) Sortieren Sie beide Teilfolgen mit Hilfe eines rekursiven Aufrufs der Methode `sort()`.
  - (c) Fügen Sie die beiden sortierten Teilfolgen zusammen mittels der zuvor implementierten Methode `union()`.
4. Bestimmen Sie die Komplexität der Methode `sort` in Abhängigkeit der Länge des Arrays `array`.