

# Übungsblatt 10

## Aufgabe 1 (Lineare Rekursion, 5 Punkte)

Implementieren Sie eine rekursive Methode, die in einem Feld mit Feldtyp `int` das kleinste Element heraussucht und zurück gibt. Das zu durchsuchende Feld soll als Parameter Ihrer Methode übergeben werden. Die Methode kann weitere Parameter besitzen.

Die Lösung soll objekt-orientiert sein. Es muss sich um lineare Rekursion handeln. Es dürfen keine Schleifen verwendet werden.

Bevor Sie mit der Implementierung anfangen, überlegen Sie sich konzeptionell die rekursive Struktur des Problems, um dann eine Idee für einen rekursiven Algorithmus zu entwickeln.

Überprüfen Sie Ihre Implementierung mit einer JUnit-Testklasse.

## Aufgabe 2 (Verzweigende Rekursion, 5 Punkte)

Implementieren Sie eine rekursive Methode `public long getBinomialKoeffizient(int n, int k)`, die den Binomialkoeffizienten mit Hilfe des Pascalschen Dreiecks berechnet:

$$\binom{n}{k} = \binom{n-1}{k-1} + \binom{n-1}{k} \text{ und } \binom{n}{0} = \binom{n}{n} = 1$$

Ihre Implementierung der Methode darf keinerlei Felder zur Speicherung von Zwischenergebnissen oder Schleifen zur Berechnung von (Teil-)Ergebnissen verwenden. Die Lösung soll objekt-orientiert sein. Testen Sie Ihre Implementierung mit Hilfe einer JUnit Testklasse.

Erweitern Sie Ihre Implementierung, so dass die Anzahl aller rekursiver Aufrufe für einen Aufruf von `getBinomialKoeffizient()` gezählt wird und mit einer Methode zurückgegeben bzw. wieder auf 0 gesetzt werden kann. Implementieren Sie eine weitere Klasse mit einer `main`-Routine, wo Sie für alle  $i$  von 0 bis 30 die Binomialkoeffizienten  $\binom{i}{\frac{i}{2}}$  berechnen und für jeden Wert von  $i$  die zugehörige Anzahl der rekursiven Aufrufe für die Berechnung des Binomialkoeffizienten auf dem Bildschirm ausgeben. Wie lange dauert die Berechnung ungefähr für  $i = 30$ , wie lange wird wohl die Berechnung für  $i = 35$  und  $i = 40$  dauern?

Vergleichen Sie die rekursive Implementierung mit der Implementierung aus der Vorlesung. Welche Vor- und Nachteile haben beide Lösungen?