

# Übungen zur Einführung in die Programmiersprache Java

Universität Regensburg  
NWF II - Physik  
Dominik Köppl

Wintersemester 2011/12  
Blatt 8

---

## 29 Weihnachtsverkauf

Sie sind stolzer Eigentümer einer Baumplantage und möchten nun vom Weihnachtsgeschäft profitieren. Dazu wollen Sie als Weihnachtsbaumverkäufer Ihren Umsatz erzielen. Auf Ihrer Plantage züchten Sie Fichten, Nordmann-Tannen und Blautannen. Bei den Blautannen hatten Sie leider bei der Aufzucht Schwierigkeiten, s.d. Ihr Angebot bei dieser Sorte sehr begrenzt ist. Sie sind nun gerade auf dem Weg zu Ihrer Plantage und wollen eine Inventur machen...

- (a) Schreiben Sie eine `Enum`, die alle Baumsorten widerspiegelt, die Sie aufzüchten.
- (b) Schreiben Sie eine Klasse `Baum` mit den Attributen `groesse` und `typ`. Wählen Sie dabei passende Datentypen und implementieren Sie für jedes Attribut je einen `Getter`. Ein `Getter` ist eine Methode, die auf ein Attribut lesend zugreift - während ein `Setter` ein Attribut manipulieren kann. Die Klasse soll zudem einen vollständigen Konvertierungs-Konstruktor verfügen. Stellen Sie sicher, dass die Attribute nach dem Setzen über dem Konstruktor nicht mehr geändert werden können und ein Direktzugriff nicht möglich ist (Stichwort `final`, `private`)
- (c) Schreiben Sie eine Klasse `Plantage`, welche ein Array von Bäumen verwaltet. Das Array hat eine feste Anfangsgröße und ist am Anfang leer. Bäume kann man mit der `void add(Baum b);`-Methode ins Array hinzufügen. Wird der Methode der `null`-Zeiger übergeben, so soll Sie die Exception `NullPointerException` werfen. Ist das Array bereits voll, soll sie die Exception `ArrayIstVoll` werfen<sup>1</sup>. Letztere Exception müssen Sie noch implementieren.
- (d) Erweitern Sie die Klasse `Plantage` um die Methode `void printInventur();`. Diese soll auswerten, wieviele Bäume welcher Art auf der Plantage stehen. Geben Sie diese Information auf dem Bildschirm aus.
- (e) Schreiben Sie eine Funktion `Baum neuerBaum();`, die mit Hilfe von Zufallszahlen einen Baum erstellt. Bedenken Sie, dass Ihre Plantage weniger Blautannen als Fichten bzw. Nordmann-Tannen beherbergt. Fügen Sie dann zufällig viele Bäume zu Ihrer Plantage hinzu und machen Sie eine Inventur.

**Auf diese Aufgabe gibt es 2 Punkte**

---

<sup>1</sup>ist auch realistisch: Eine Baumschule kann nur begrenzt viele Bäume aufnehmen

## 30 Große Zahlen

Berechnen Sie die Fibonacci-Folge nun *iterativ* mit dem Datentyp `BigInteger`, um auch größere Folgenglieder schnell abfragen zu können. Schreiben Sie dazu eine Funktion

```
BigInteger fib(int n)
```

und überprüfen Sie das Ergebnis

```
fib1000 = 434665576869374564356885276750406258025646605173717804024817290895365-  
554179490518904038798400792551692959225930803226347752096896232398733224711616-  
42996440906533187938298969649928516003704476137795166849228875
```

## 31 Die Koch'sche Schneeflocke

Das Finale unserer Arbeit mit der `pencil`-Klasse bietet die Koch'sche Schneeflocke.

- Eine Koch'sche Schneeflocke ist aus mehreren Koch-Kurven zusammengesetzt.
- Die Koch-Kurve ist rekursiv definiert.
- Eine Koch-Kurve nullter Ordnung ist eine Strecke.
- Eine Koch-Kurve  $n$ .ter Ordnung besteht aus vier Streckenabschnitten
  - Koch-Kurve  $n - 1$ -ter Ordnung mit  $\frac{1}{3}$  der Strecke
  - Drehung um 60 Grad gegen Uhrzeigersinn
  - Koch-Kurve  $n - 1$ -ter Ordnung mit  $\frac{1}{3}$  der Strecke
  - Drehung um 120 Grad im Uhrzeigersinn
  - Koch-Kurve  $n - 1$ -ter Ordnung mit  $\frac{1}{3}$  der Strecke
  - Drehung um 60 Grad gegen Uhrzeigersinn
  - Koch-Kurve  $n - 1$ -ter Ordnung mit  $\frac{1}{3}$  der Strecke
- Wir drehen nach jeder Zeichnung der Koch-Kurve den `Pencil` um 120 Grad und zeichnen erneut eine Koch-Kurve.
- $\Rightarrow$  Wir erhalten eine Schneeflocke.

## 32 Satzzeichen

Schreiben Sie eine Funktion `zaehleSatzzeichen`, die einen `String str` als Parameter erhält und eine Integervariable zurückgibt. Die Funktion soll alle vorkommenden Satzzeichen in `str` aufzählen und diese Anzahl zurückgeben. Als Satzzeichen kommen Ausrufezeichen, Fragezeichen, Komma und Punkt in Frage. Verwenden Sie dazu die Methode `char charAt(int index)` der Klasse `String`. Um eine `NullPointerException` zu verhindern, muss der übergebene Parameter geprüft und ggf. der Standardwert 0 zurückgegeben werden. Ein Funktionsaufruf von `zaehleSatzzeichen("Ich und Du, Müller's Kuh. Müller's Esel, der bist Du!")`; würde den Wert "vier" zurückliefern.