

Übungsblatt 03

Zu bearbeiten bis 01.12.2005

Aufgabe 1

Im Übungsblatt 02 haben Sie einen Algorithmus entwickelt, Geldbeträge bis 4,99 € mit möglichst großen Münzen auszuführen. Implementieren Sie den Algorithmus in Java unter Verwendung des Rahmenprogrammes [Euro.java](#). Das Rahmenprogramm liest den Geldbetrag in ct als Kommandozeilenparameter ein (Aufruf z.B. `java Euro 407`) und soll dann ausgeben $2*200+1*5+1*2$)



Aufgabe 2

Der folgende Algorithmus berechnet die Summe der Zahlen von 1 bis n, für $n > 0$.

```
sum=0;
i=0;
while (i<n) {
  // hier gilt:  $sum = \sum_{j=1}^i j \wedge i \leq n$ 
  i=i+1;
  sum=sum+i;
  // hier gilt:  $sum = \sum_{j=1}^i j \wedge i \leq n$ 
}
// Schlussbedingung:  $sum = \sum_{j=1}^n j$ 
```

- Zeigen Sie, dass $sum = \sum_{j=1}^i j \wedge i \leq n$ Schleifeninvariante ist (es muss kein mathematisch exakter Beweis sein).
- Zeigen Sie, dass der Algorithmus partiell korrekt ist, indem Sie aus der Schleifeninvariante auf die Schlussbedingung schließen.
- Implementieren Sie eine Methode `public static int sum(int n)`, die die Summe der Zahlen von 1..n wie oben berechnet.
- Schreiben Sie eine Testklasse `public class summe`, deren main-Methode die sum-Methode für $n = 1, \dots, 100$ aufruft.

- e) Überlegen Sie, welches das größte n ist, für das dieses Java-Programm funktioniert (der Wertebereich des Java-Typs `int` endet bei `Integer.MAX_VALUE`).

Aufgabe 3

Gegeben sei folgende rekursiv definierte Funktion f :

$$f(1) = 1$$

$$f(n) = f(n-1) + 2n - 1$$

- Was berechnet $f(n)$?
- Implementieren Sie diese rekursive Funktion in Java als statische Methode `static int square(int n)` in einer Testklasse `Quad`, deren `main`-Methode $f(1)$ bis $f(20)$ berechnet und ausgibt.
- Implementieren Sie alternativ eine statische Methode `static int itsqaure(int n)`, die die Funktion f iterativ nach obiger Vorschrift berechnet.