



## Programming by Contract

SS 2007 – Übungsblatt 12

Ausgabe: 12. Juli 2007  
(optional)

### **Aufgabe 1.** *Subcontracting*

Konzipieren Sie für die Elter- und Kindklasse Kunde/Stammkunde eine (C++-)virtuelle Methode, die das Vorgehen beim Subcontracting veranschaulicht (konkretes Beispiel): Welche Klasse ist die Sub-, welche die Superklasse? Was ist der Contract der Superklasse? Wann wird dieser Contract erfüllt, wann nicht? ...

### **Aufgabe 2.** *Methodenklassifizierung*

Konzipieren Sie eine Klasse `rationalNumber` und benutzen Sie dabei die Einteilung *basic queries*, *invariant*, *derived queries*, *constructors*, *destructor*, *assignment operator*, *operator+*, *operator==*, *modifiers*.

Schreiben Sie für die Konstruktoren, den `operator+`, den `Operator==` und einen weiteren arithmetischen Operator Ihrer Wahl einen vollständigen Contract in Nana-Syntax. Vergessen Sie nicht eine geeignete Invariante!

### **Aufgabe 3.** *Frameregeln*

Was ist eine Frameregeln?

Stellen Sie die in der Vorlesung besprochenen alternativen Möglichkeiten der Frameregeln-Spezifikation der Klassen `mydirectory`, `name_list`, ... gegenüber und diskutieren Sie jeweils Vor- sowie Nachteile.

**Aufgabe 4.** *Konstruktoren/Destruktoren*

Schreiben Sie je eine „Leitlinie“, wie C++-Konstruktoren und wie C++-Destruktoren in Nana-Syntax spezifiziert werden sollten.

**Aufgabe 5.** *operator=*

Spezifizieren Sie einen Wertzuweisungsoperator

```
template <class KEY, class VALUE>
mydictionary<KEY, VALUE>& mydictionary<KEY, VALUE>::operator=(
    const mydictionary<KEY, VALUE>& s)
{
    //...
};
```

für die Klasse mydictionary.