

## Übungsblatt 4

Abgabe bis Freitag, 23.11.2012, 12:00 Uhr

### Hinweis:

Lösungen immer per Email an den zuständigen Tutor schicken. Die Emailadressen sind auf der Homepage zur Vorlesung gelistet.

### Aufgabe 4.1

In dieser Aufgabe soll eine Klasse `Vector2` implementiert werden, die Vektoren  $(x, y)$  im  $\mathbb{R}^2$  repräsentiert.

1. Implementieren Sie die Klasse `Vector2` mit einem Konstruktor, der es erlaubt, die Werte des Vektors zu setzen.
2. Implementieren Sie Methoden zum Setzen und Auslesen der Werte mit den folgenden Prototypen:

```
void setX(double x);  
void setY(double y);  
double x();  
double y();
```

3. Implementieren Sie eine Methode `toString()`, die den Vektor im Format  $(x, y)$  als String zurückgibt.
4. Implementieren Sie einen Konstruktor `Vector2(Vector2 o)`, der ein Objekt der Klasse `Vector2` als Parameter erwartet und die  $x$ - und  $y$ -Werte dieses Objekts übernimmt.

### Aufgabe 4.2

1. Implementieren Sie eine Klasse `Calculator`, die Vektorrechnungen im  $\mathbb{R}^2$  mit der Klasse `Vector2` ausführt.
  - Implementieren Sie eine Methode `multiply`, die ein `double` und ein `Vector2` Objekt als Parameter erwartet. Die Methode soll den `Vector2` mit einem `double` multiplizieren und das Ergebnis zurückgeben.
  - Implementieren Sie eine Methode `add`, die zwei `Vector2` Objekte als Parameter erwartet. Die Methode soll die beiden `Vector2` addieren und das Ergebnis zurückgeben.

- Implementieren Sie eine weitere Methode, die drei Vektoren addiert und das Ergebnis zurückgibt. Überladen Sie hierzu die Methode `add`.
- Erweitern Sie die Klasse `Calculator` so, dass nach den Rechenoperationen `multiply` und `add` das Ergebnis in einer Instanzvariablen `ans` gespeichert wird. Schreiben Sie eine Methode `ans`, die das Ergebnis der letzten Rechenoperation zurückgibt.
- Implementieren Sie eine Methode `dot`, die zwei `Vector2` Objekte als Parameter erwartet und das Skalarprodukt der zwei Vektoren im Euklidischen Raum berechnet. Die Methode soll ein `double` zurückgeben.

2. Diskutieren Sie, warum der folgende Programmcode Ergebnisse ausgibt, die mathematisch nicht korrekt sind.

```
public class Numbers {
    public static void main(String args[]) {

        double result1 = 1 / 2 * 3; // Zeile 3
        System.out.println("1 / 2 * 3 = " + result1);

        double g = 10;
        double h = 1 / g ;
        double result2 = h * h * h * g * g * g - 1.0; // Zeile 8
        System.out.println("0.1^3 * 10^3 - 1 = " + result2);
    }
}
```