

Programmieren WS 2011/2012

http://zvi.ipd.kit.edu/lehre_programmieren_ws11.php

Dozent: Prof. Dr. A. Pretschner
Organisation: Florian Kelbert kelbert@kit.edu
Jürgen Graf graf@kit.edu
Martin Mohr martin.mohr@kit.edu

Tutorium 7

05.12.2011 - 09.12.2011

Aufgabe 1 (Liste von Punkten: **LinkedList**)

Implementieren Sie die Klasse **LinkedList**, die Punkte (Klasse **Point**) mittels einer Liste verwaltet. Schreiben Sie hierzu die für Listen üblichen Schnittstellen:

```
public class LinkedList {  
    public void add(Point p, int index) { ... }  
    public void add(Point p) { ... }  
    public void addFirst(Point p) { ... }  
    public void addLast(Point p) { ... }  
    public boolean contains(Point p) { ... }  
    public Point get(int index) { ... }  
    public Point getFirst() { ... }  
    public Point getLast() { ... }  
    public int indexOf(Point p) { ... }  
    public boolean isEmpty() { ... }  
    public void remove(int index) { ... }  
    public void remove(Point p) { ... }  
    public void removeFirst() { ... }  
    public void removeLast() { ... }  
    public int size() { ... }  
}
```

Verwenden Sie hierzu die folgende Klasse **Point**:

```
public class Point {  
    private double x;  
    private double y;  
  
    public Point(double x, double y) {  
        this.x = x;  
        this.y = y;  
    }  
  
    public double getX() {  
        return this.x;  
    }  
}
```

```
    }

    public double getY() {
        return this.y;
    }

    public boolean equals(Point p) {
        return this.x == p.x &&
               this.y == p.y;
    }
}
```

Aufgabe 2 (Sortieren der `LinkedList`)

Schreiben Sie in der Klasse `LinkedList` die Methode

```
public void sortByDistanceToOrigin(),
```

die die Liste so sortiert, dass die enthaltenen Punkte anschließend **aufsteigend** nach ihrem Abstand zum Ursprung sortiert sind. Für einen Punkt mit Koordinaten x und y , (x, y) , berechnet sich dabei der Abstand d zum Ursprung als

$$d = \sqrt{x^2 + y^2}$$