

# Übungsblatt 6

(3. Mai 2010)

**Aufgabe 1** Die Unterklasse von `GeometricObject`, die rotierende Objekte darstellt aus dem letzten Übungsblatt ist eine unglückliche Modellierung. Wir können in dieser Modellierung nicht eine Klasse für rotierende Sterne ableiten, die sowohl Unterklasse der Sternklasse als auch Unterklasse der Klasse für rotierende Objekte ist. Wir können in Java nur von einer Klasse erben.

Daher sollen Sie in dieser Aufgabe eine andere Modellierung wählen. Gegeben sei folgende Schnittstelle:

```

1 package name.panitz;
2
3 public interface Mover{
4     void move(GeometricObject geo);
5     void reverse();
6 }
```

- a) Lassen Sie jetzt Ihre Klasse `GeometricObject` ein Feld des Typs `Mover` haben. In der Methode `move` von `GeometricObject` soll das `Mover`-Objekt das `GameObject` weiterbewegen.
- b) Implementieren Sie eine `Mover`-Klasse, die eine lineare Bewegung realisiert.
- c) Implementieren Sie eine `Mover`-Klasse, die eine Kreisbewegung realisiert.
- d) Implementieren Sie eine weitere `Mover`-Klasse, die eine Bewegung Ihrer Wahl implementiert.

**Aufgabe 2** Implementieren Sie in den Klassen `Li` und `ArrayList` aus der Vorlesung vom 29. April die fehlenden Listenmethoden aus dem Listen-Interface `List` mit Ausnahme der Methode `sort`. Überlegen Sie dabei, welche Methoden in der abstrakten Klasse `AbstractList` implementiert werden könnten.