

# Informatik I: Einführung in die Programmierung

Prof. Dr. Bernhard Nebel  
Dr. Stefan Wöfl, Thorsten Engesser  
Wintersemester 2015/2016

Universität Freiburg  
Institut für Informatik

## Übungsblatt 10

**Abgabe: Freitag, 15. Januar 2016, 20:00 Uhr**

**WICHTIGE HINWEISE:** Zur Bearbeitung der Übungsaufgaben legen Sie bitte ein neues Unterverzeichnis `sheet10` im Wurzelverzeichnis Ihrer Arbeitskopie des SVN-Repositories an. Ihre Lösungen werden dann in Dateien in diesem Unterverzeichnis erwartet.

Beachten Sie bitte bei allen Aufgaben die *Hinweise zur Bearbeitung der Übungsaufgaben* unter der folgenden URL:

<http://gki.informatik.uni-freiburg.de/teaching/ws1516/info1/wiki/hinweise.html>

Bewertet wird bei allen Aufgaben die letzte Version, die zur Deadline des Übungsblattes auf dem SVN-Server eingereicht ist.

**Aufgabe 10.1** (MRO, 4 Punkte, Datei: `mro.txt`)

Betrachten Sie die folgenden Klassendefinitionen:

```
class A:
    pass

class B:
    pass

class C:
    pass

class D(A, B):
    pass

class E(B, C):
    pass

class F(D, E):
    pass
```

Wenden Sie auf diese Klassenhierarchie den im Foliensatz 21 beschriebenen C3-Algorithmus an, um die Method-Resolution-Order für die Klasse `F` zu bestimmen. Geben Sie alle Zwischenschritte an.

**Aufgabe 10.2** (RoboRally, Punkte 2+3+3, Datei: `hw_roborally.py`)

In dieser Aufgabe beschäftigen wir uns mit der Implementierung weiterer Fabrikelemente für das in der Vorlesung vorgestellte Brettspiel *RoboRally*. Erweitern Sie dazu die Datei `hw_roborally.py`, die Sie von der Vorlesungswebseite herunterladen können:

[http://www.informatik.uni-freiburg.de/~ki/teaching/ws1516/info1/python/hw\\_roborally.py](http://www.informatik.uni-freiburg.de/~ki/teaching/ws1516/info1/python/hw_roborally.py)

In dieser Datei sind die Klassen und die zu bestehenden Doctests bereits angegeben. Ihre konkrete Aufgabe ist es, die beiden Fabrikelemente „Reparaturfeld“ und „Einweg-Portal“ als neue Klassen zu implementieren:

**Reparaturfeld (RepairSite):** Falls ein Roboter zu Ende einer Registerphase auf einem Reparaturfeld zum Stehen kommt, wird ihm je nach Konfiguration des Reparaturfelds ein oder zwei Schadenspunkte abgezogen. Dies geschieht im 8. Arbeitsschritt.

**Einweg-Portal (OneWayPortal):** Ein Einweg-Portal schickt einen Roboter von seiner momentanen Position zu einer anderen, bei der Initialisierung des Portals angegebenen, Position. Dies geschieht im 6. Arbeitsschritt. Bei Kollisionen wird ähnlich wie bei Förderbändern vorgegangen: Alle Roboter werden parallel bewegt. Existiert am Ende eine Kollision, werden die Bewegungen der kollidierten Roboter zurück abgewickelt. Beachten Sie mögliche unerwünschte Interaktionen mit Wänden!

Die Aufgaben:

- (a) Überlegen und begründen Sie, wie die zwei neuen Fabrikelemente in die bestehende Klassenhierarchie (siehe auch Folie 32 der Vorlesung `infoI21.pdf`) einzuordnen sind. Wovon sollten `RepairSite` und `OneWayPortal` jeweils sinnvoller Weise abgeleitet werden und warum?
- (b) Implementieren Sie die Klasse `RepairSite`, die das oben beschriebene „Reparaturfeld“ realisiert. Die genaue Verwendung wird aus den Doctests ersichtlich.
- (c) Implementieren Sie die Klasse `OneWayPortal`, die das oben beschriebene „Einweg-Portal“ realisiert. Siehe auch hier die entsprechenden Doctests.

**Aufgabe 10.3** (Suchbäume und Klassen; Punkte: 6; Datei: `tree.py`)

Schreiben Sie eine Klasse `SearchTree`, die einen Suchbaum implementiert (siehe Foliensatz 11). Sie können davon ausgehen, dass in dem Suchbaum nur Objekte enthalten sind (bzw. hinzugefügt werden), die sich durch die Vergleichsoperatoren `<` und `==` ordnen lassen. Ihre Klasse soll die folgenden Methoden enthalten:

- `__init__(self, items=None)`  
Initialisiert den neuen Suchbaum mit den Elementen aus `items` (z.B. eine Liste).
- `__contains__(self, item)`  
Gibt `True` zurück, wenn sich das Element `item` im Suchbaum befindet, sonst `False`.
- `add(self, item)`  
Fügt das Element `item` in den Suchbaum ein. Befindet sich bereits ein gleiches Element im Suchbaum, soll der Baum nicht verändert werden.
- `tolist(self)`  
Gibt eine Liste mit den Elementen des Suchbaums in In-Order-Reihenfolge zurück.

Schreiben Sie für jede Methode eine passende Testfunktion.

**Aufgabe 10.4** (Erfahrungen; Datei: `erfahrungen.txt`; Punkte: 2)

Legen Sie im Unterverzeichnis `sheet10` eine Textdatei `erfahrungen.txt` an. Notieren Sie in dieser Datei kurz Ihre Erfahrungen beim Bearbeiten der Übungsaufgaben (Probleme, benötigter Zeitaufwand nach Teilaufgabe, Bezug zur Vorlesung, Interessantes, etc.).