

Informatik I: Einführung in die Programmierung

Prof. Dr. Bernhard Nebel
Tim Schulte, Thorsten Engesser
Wintersemester 2017/2018

Universität Freiburg
Institut für Informatik

Übungsblatt 11

Abgabe: Freitag, 19. Januar 2018, 20:00 Uhr

WICHTIGE HINWEISE: Zur Bearbeitung der Übungsaufgaben legen Sie bitte ein neues Unterverzeichnis `sheet11` 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/ws1718/info1/guide/hinweise.html>

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

Aufgabe 11.1 (RoboRally, Punkte 2+5+5, Datei: `roborally.txt`, `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. In dieser Datei sind die Klassen und die zu bestehenden Doctests bereits angegeben. Ihre konkrete Aufgabe ist es, die beiden Fabrikelemente „Reparaturfeld“ (`RepairSite`) und „Teleporter“ 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.

Teleporter: Ein Teleporter schickt einen Roboter von seiner momentanen Position zu einer, bei der Initialisierung des Teleporters angegebenen, Zielposition. Dies geschieht im 7. Arbeitsschritt. Ist die Zielposition bereits von einem anderen Roboter belegt, tauschen diese ihre Plätze.

Die Aufgaben:

- 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 `Teleporter` jeweils sinnvoller Weise abgeleitet werden und warum? Speichern Sie Ihre Antworten in der Datei `roborally.txt`.
- Implementieren Sie die Klasse `RepairSite`, die das oben beschriebene „Reparaturfeld“ realisiert. Die genaue Verwendung wird aus den Doctests ersichtlich.
- Implementieren Sie die Klasse `Teleporter`, die das oben beschriebene Fabrikelement realisiert. Siehe auch hier die entsprechenden Doctests.

Aufgabe 11.2 (MRO, 6 Punkte, Datei: `mro.txt`)

Betrachten Sie die folgenden Klassendefinitionen:

```
class A:
    pass

class B:
    pass

class C(A):
    pass

class D(A, B):
    pass

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

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

Aufgabe 11.3 (Erfahrungen; Datei: `erfahrungen.txt`; Punkte: 2)

Legen Sie im Unterverzeichnis `sheet11` 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.).