

# Funktionale und objektorientierte Programmierkonzepte

## Übungsblatt 00



TECHNISCHE  
UNIVERSITÄT  
DARMSTADT

### Entwurf

**Achtung:** Dieses Dokument ist ein Entwurf und ist noch nicht zur Bearbeitung/Abgabe freigegeben. Es kann zu Änderungen kommen, die für die Abgabe relevant sind. Es ist möglich, dass sich **alle** Aufgaben noch grundlegend ändern. Es gibt keine Garantie, dass die Aufgaben auch in der endgültigen Version überhaupt noch vorkommen und es wird keine Rücksicht auf bereits abgegebene Lösungen genommen, die nicht die Vorgaben der endgültigen Version erfüllen.

### Hausübung 00

**Gesamt: 5 Punkte**

#### Erste Praxis mit Java & FopBot

Beachten Sie die Seite *Verbindliche Anforderungen für alle Abgaben im Moodle-Kurs*.

Verstöße gegen verbindliche Anforderungen führen zu Punktabzügen und können die korrekte Bewertung Ihrer Abgabe beeinflussen. Sofern vorhanden, müssen die in der Vorlage mit TODO markierten crash-Aufrufe entfernt werden. Andernfalls wird die jeweilige Aufgabe nicht bewertet.

Die für diese Hausübung relevanten Verzeichnisse sind `src/main/java/h00` und ggf. `src/test/java/h00`.

### H1: Matrikelnummer in Moodle

**?? Punkte**

In dieser Aufgabe erhalten Sie einen Punkt dafür, dass Sie Ihre Matrikelnummer *fehlerfrei* in Moodle eintragen oder bereits eingetragen haben. Wenn Sie Ihre Matrikelnummer bereits in Moodle eingetragen haben, vergewissern Sie sich *dringend*, dass diese korrekt ist!

#### Hinweis:

In unserem Studierenden-Guide finden Sie im Abschnitt *Vorbereitung* → *Eintragen Ihrer Matrikelnummer* eine Anleitung dazu, wie Sie Ihre Matrikelnummer in Moodle eintragen.

Wenn Sie Ihre Matrikelnummer nicht oder nicht korrekt in Moodle hinterlegt haben, kann Ihre Zulassung und Ihr Bonus unter Umständen nicht korrekt verbucht werden!

### H2: Einrichten von Java und Ihrer Entwicklungsumgebung

**?? Punkte**

Zum Bearbeiten unserer Hausübungen verwenden Sie eine Java-Entwicklungsumgebung. Hierzu zählt zum Beispiel *IntelliJ*: Die Verwendung von IntelliJ wird von uns *dringend* empfohlen und unterstützt.

#### Hinweis:

In unserem Studierenden-Guide finden Sie im Abschnitt *Vorbereitung* → *Installieren von Java* und *Vorbereitung* → *Installieren von IntelliJ* Guides zur Installation von Java bzw. IntelliJ.

---

### H3: Herunterladen und Importieren unserer Vorlagen

**?? Punkte**

In der FOP stellen wir Ihnen für *jede* Hausübung eine Vorlage bereit, die zur Bearbeitung der jeweiligen Hausübung verwendet werden muss.

**Hinweis:**

In unserem Studierenden-Guide finden Sie im Abschnitt *Vorbereitung* → *Installieren von Java* und *Vorbereitung* → *Installieren von IntelliJ* Guides zur Installation von Java bzw. IntelliJ.

Beachten Sie weiter die auf jedem Übungsblatt am Anfang erwähnte Seite *Verbindliche Anforderungen für alle Abgaben* sowie die mitgelieferten Hinweise.

---

### Einleitung: Geschwisterliebe

*Im Zimmer, sie teilen es zu zweit,  
Alfred und Kaspar, Geschwister in der Zeit.  
Kaspar streut Münzen, macht das Chaos groß,  
Alfred hebt sie auf, bleibt dabei nicht bloß.*

*Doch Kaspar spielt weiter, sein böses Spiel,  
Münzen verlegt er, wild und viel.  
Alfred ist müde, von der Arbeit erschöpft,  
Athletisch sein, wie er es gern möchte’.*

*Der Zorn packt ihn, wild tanzt er nun,  
Alfred lacht, hat seinen Spaß daran gewun’n.  
Im Zimmer ein Tanz, voller Wut und Gier,  
Geschwisterliebe versteckt sich hier.*

Mit einem ungewöhnlichen Einstieg hat man sich den Spaß erlaubt dieses Übungsblatt mit einem feinen Gedicht das Szenario darzustellen, die wir selber in unserer 'FopBot' - Welt mit Java zu implementieren haben - keine Sorge, eine Gedichtsanalyse steht hier nicht auf der Agenda. ☺

Alfred und Kaspar sind zwei Geschwister, die sich in der 'FopBot' - Welt (= gemeinsames Zimmer der beiden) aufhalten. Hierbei sind beide Roboter charakteristisch gesehen Gegensätze, da Kaspar ein Roboter ist, der gerne Unordentlichkeit stiftet und seine Münzen in der Welt verteilen mag.

Alfred hingegen ist ein disziplinierter Roboter, der die 'FopBot' - Welt sauber und ordentlich halten möchte. Wann immer Kaspar seine Unruhe stiftet, fühlt sich Alfred gezwungen entgegen zu wirken, sprich er will und soll die Münzen aufsammeln. Jedoch soll angenommen werden, dass Alfred nicht so flink und ausdauernd ist wie der Kaspar und deswegen sehr schnell überfordert werden kann.

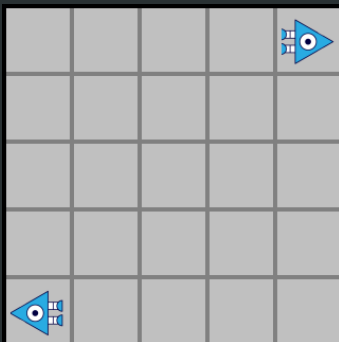
Wir wollen nun versuchen, diesen Akt nachzuahmen...

**H4: Kaspar, der Zimmerchaot****?? Punkte**

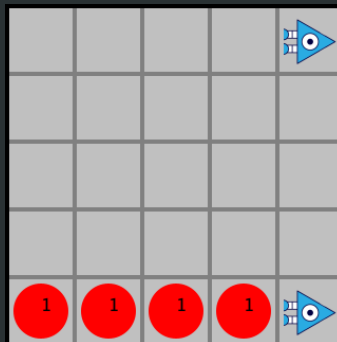
Kaspar soll zuerst damit anfangen, seine Münzen in der Welt abzulegen und zwar so, dass das endgültige Muster einem spiegelverkehrten 'L'-Buchstaben ähnelt. Dafür soll er sich als erstes so oft drehen, dass er nach Osten schaut. Nun soll er sich der unteren Wand **entlang** vorwärts bewegen, wobei er **vor** jedem Vorwärtsschritt **eine** Münze ablegen soll.

An der unteren rechten Ecke der Welt angekommen soll er sich einmal nach links drehen und seinen Weg zur oberen rechten Ecke der Welt fortsetzen und die **gleiche Vorgehensweise** wie gerade geschildert **wiederholen**, heißt sich nach vorne bewegen und davor auch eine Münze ablegen.

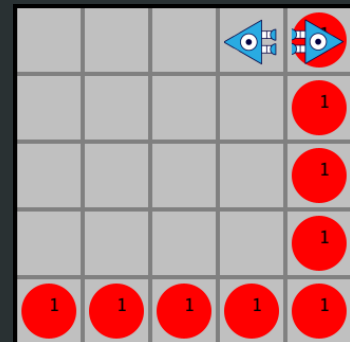
Anschließend an der oberen rechten Ecke angekommen, wo der Alfred platziert ist, soll der Kaspar sich einmal nach links drehen, eine Münze ablegen und einen Vorwärtsschritt machen (wir wollen den armen Alfred nicht 'verdecken'), wodurch am Ende auch das spielverkehrte 'L'-Münzenmuster erreicht werden soll.



- (a) Anfangszustand der Welt:
- Roboter 'Kaspar'(0,0) = Spalte 0, Zeile 0 bzw. unten links, Blickrichtung **Westen**;
  - Roboter 'Alfred'(4,4) = Spalte 4, Zeile 4 oben rechts, Blickrichtung **Osten**



- (b) Zwischenzustand der Welt nach Erreichen der rechten unteren Ecke der Welt:
- Roboter 'Kaspar'(4,0) = unten rechts;
  - Roboter 'Alfred'(4,4) = oben rechts



- (c) Endzustand der Welt nach der H4:
- Roboter 'Alfred'(4,4) = oben rechts
  - Roboter 'Kaspar'(3,4) = links von Alfred

Fangen Sie nun damit an die Reihe von Anweisungen, wie oben geschildert, zu schreiben, die der Roboter 'Kaspar' ausführen soll. Beachten Sie, dass in der FopBot-Welt zwei Roboter mit unterschiedlicher Position und Blickrichtung positioniert wurden. Anhand der Bezeichner der Roboter sollte Ihnen klar sein, welchen Roboter Sie anzusprechen und die Methoden anzuwenden haben.

**Hinweis:**

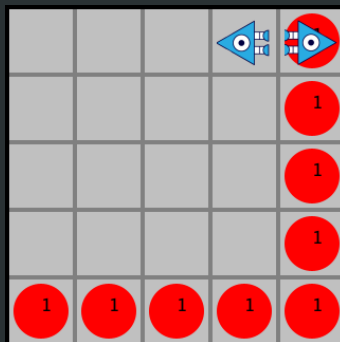
- Sie dürfen in *dieser* Hausübung davon ausgehen, dass die Welt immer die Größe 5x5 besitzt. Somit sollten Sie herleiten können, wie viele Vorwärtsschritte (und wann) in den verschiedenen Szenarien auszuführen sind.
- Schreiben Sie Ihren Code an der designierten **'TODO'**-Stelle in der Datei **Main.java** in **src/main/java/h00** (Sie werden nur in dieser Datei Ihre Abgaben schreiben müssen).
- Vergessen Sie nicht, den Aufruf der Methode **'crash()'** zu beseitigen, falls Sie Ihren Code ausführen bzw. testen wollen (ohne, dass am Ende das Programm abstürzt und eine Fehlermeldung wiedergegeben wird).

**H5: Alfred, der Ordnungsfanatiker****?? Punkte**

Nun ist es an der Reihe, den **Alfred** in Bewegung zu bringen. Er soll im Prinzip das wiederkehren, was sein Bruder angestiftet hat. Das heißt er soll vor jeder Vorwärtsbewegung die Münze **aufheben** und somit auch am Ende das Münzenmuster beseitigt haben.

Wenn er in der unteren linken Ecke der Welt angekommen ist, versuche ihn um einen Schritt **nördlich** zu platzieren.

Der Alfred ist anfangs oben rechts in der 'FopBot'-Welt platziert und blickt, wie es anhand der Abbildung zu entnehmen ist, in Richtung **Osten**.



(a) Anfangszustand der Welt bzw. Endzustand nach der H4:  
 - Roboter 'Kaspar'(3,4), Blickrichtung **Westen**;  
 - Roboter 'Alfred'(4,4) = oben rechts, Blickrichtung **Osten**



(b) Endzustand der Welt nach der H5:  
 - Roboter 'Kaspar'(3,4);  
 - Roboter 'Alfred'(1,0)

**Verbindliche Anforderungen:**

Sie haben in dieser Aufgabe von der Nutzung von **'for'**-Schleifen Gebrauch zu machen, falls Sie es noch nicht schon in der vorherigen Aufgabe gemacht haben sollten. Das Bewegungsmuster in der H4 (1x Münze ablegen - 1x Vorwärtsschritt) musste mal oftmals durchführen und sollte auch in dieser Aufgabe leicht modifiziert (wir heben diesmal die Münze auf) der Fall sein. Und für solche Situationen ist die Verwendung von Schleifen ein Segen zur Reduzierung von Schreibarbeit.

Überlegen Sie, an welchen Stellen die Verwendung von **'for'**-Schleifen in Frage kommen könnte. Und wie oft haben diese Schleifen ihre umfassenden Anweisungen (-> welche und in welcher Reihenfolge) auszuführen ?

**H6: Let's dance!****?? Punkte**

Zum großen Abschluss soll Kaspar, so frech wie er ist, seine restlichen Münzen ablegen, um Alfred (wieder) aus der Fassung zu bringen. Hierbei 'tanzt' bzw. dreht sich Alfred nach links, **solange** der Kaspar noch Münzen hat.

Anders formuliert: nach jeder Platzierung einer Münze vom Rest durch Kaspar soll Alfred eine Linksdrehung durchführen, und zwar solange der Kaspar noch Münzen hat. Die Ablegung der Münzen erfolgt dort, wo Kaspar zuletzt positioniert war (siehe Abbildung (b) von der H5).

**Verbindliche Anforderungen:**

Neben einer **'for'**-Schleife existiert auch die sog. **'while'**-Schleife.  
 Bearbeiten Sie die Aufgabe mit einer bloßen **'while'**-Schleife.