

# Funktionale und objektorientierte Programmierkonzepte

## Übungsblatt 09



TECHNISCHE  
UNIVERSITÄT  
DARMSTADT

Prof. Karsten Weihe

Wintersemester 23/24

Themen:

Relevante Foliensätze:

Abgabe der Hausübung:

v1.0

<Themen>

<1>

XX.XX.202X bis 23:50 Uhr

**Hausübung 09**

<Übungstitel>

**Gesamt: 19 Punkte**

**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/h09` und ggf. `src/test/java/h09`.

**Einleitung**

---

**H1: <Aufgabentitel>****?? Punkte**

---

&lt;Aufgabentext&gt;

---

**H1.1:****?? Punkte**

---

Überführen Sie die gegebene Klasse `StackOfObjects` in eine generische Klasse.

Die Klasse `StackOfObjects` soll einen unbeschränkten Typparameter `T` haben. Weiter soll der erste formale Parameter der Objektmethode `push` auf `T` und Subtypen `T` beschränkt sein und der Rückgabetypp der Objektmethode `pop` gleich `T` sein.

---

**H1.2:****?? Punkte**

---

Erstellen Sie im Package `h09.stack` eine generische `public`-Klasse `StackOfNumbers`, welche einen Typparameter `T` hat und direkt von der Klasse `StackOfObjects` abgeleitet ist. Der Typparameter `T` ist auf den Typ `Number` und Subtypen von `Number` beschränkt. Der Typparameter `T` der Klasse `StackOfObjects` wird mit `T` instanziiert.

---

**H2: Operationen****?? Punkte**

---

---

**H2.1:****?? Punkte**

---

Überführen Sie die rückgabelose Klassenmethode `filter` in eine generische Klassenmethode mit einem Typparameter `T`.

Instanziiieren Sie die Typparameter der formalen Parameter so, dass (1) der erste aktuelle Parameter ein beliebiger Stack sein kann, aus welchem Objekte des Typs `T` gelesen werden können, (2) der zweite aktuelle Parameter ein beliebiger Stack sein kann, in welchen Objekte des Typs `T` geschrieben werden können und (3) der dritte aktuelle Parameter ein beliebiger Filter sein kann, welcher auf Objekte des Typs `T` angewendet werden kann. Passen Sie die innerhalb der Methode verwendeten Typen entsprechend an.

---

**H2.2:****?? Punkte**

---

Überführen Sie die rückgabelose Klassenmethode `map` in eine generische Klassenmethode mit zwei Typparametern `O` und `I`.

Instanziiieren Sie die Typparameter der formalen Parameter so, dass (1) der erste aktuelle Parameter ein beliebiger Stack sein kann, aus welchem Objekte des Typs `O` gelesen werden können, (2) der zweite aktuelle Parameter ein beliebiger Stack sein kann, in welchen Objekte des Typs `I` geschrieben werden können und (3) der dritte aktuelle Parameter eine beliebige Funktion sein kann, welche ein Objekt des Typs `O` auf ein Objekt des Typs `I` abbilden kann. Passen Sie die innerhalb der Methode verwendeten Typen entsprechend an.

---

**H3: Testen mittels JUnit****?? Punkte**

---

TODO Snippet, dass Tests in Verzeichnis test erstellt werden müssen.

---

**H3.1: Test von filter****?? Punkte**

---

X

---

**H3.2: Test von map****?? Punkte**

---

X