

# 5. Übungsblatt

**Ausgabe:** 22.11.21

## 5.1 *Mehr Listen in Java*

In der Vorlesung wurden Listen in Java vorgestellt, mit den Funktionen höherer Ordnung `filter` und `map`. Ändern Sie die Implementierung von `map` so ab, dass sie korrekt über zwei Typen parametrisiert ist.

Schreiben Sie außerdem einige Tests (in der `main`-Methode), die die Benutzung von `map` und `filter` zeigen. Sie können dazu anonyme Funktionen in Java (Lambda) verwenden.

## 5.2 *Methodenaufrufe in Python*

Die Beispieldatei `methodcall.java` zeigte in der Vorlesung, wie Subtyping, Überladen und dynamisches Binden in Java interagieren.

1. Wie sieht das Beispiel direkt übersetzt in Python aus, und was passiert? Was wird dynamisch aufgelöst, was wird statisch aufgelöst?
2. Wie müsste man das Beispiel erweitern, um die gleiche Ausgabe wie in Java zu erreichen?

## 5.3 *Mehr Dynamik*

In Haskell werden Funktionsaufrufe statisch (zur Übersetzungszeit) aufgelöst. Wie würde man dynamisches Binden in Haskell implementieren?