

# Erweiterte Grundlagen der Programmierung

Dr. Marcel Lüthi  
Herbstsemester 2020

Universität Basel  
Fachbereich Informatik

## Hinweise zu den Tests

Für viele Übungsaufgaben der Veranstaltung *Erweiterte Grundlagen der Programmierung* werden automatische Testprogramme mitgeliefert. Hier finden Sie eine Anleitung, wie diese Tests verwendet werden können.

*Am besten lesen Sie sich dieses Blatt gründlich durch, bevor Sie das erste Übungsblatt lösen, aber nachdem Sie dieses heruntergeladen und die Zip-Datei entpackt haben. Damit können Sie die Anweisungen gleich ausprobieren.*

**Automatisierte Tests** Ein Test hilft Ihnen dabei festzustellen, ob ihr Programm richtig funktioniert oder ob es noch versteckte Fehler enthält, die Sie möglicherweise noch nicht entdeckt haben. Ein Test besteht aus mehreren kleinen Prüfprogrammen, die das von Ihnen entwickelte Programm auf Richtigkeit überprüft. Falls ein Test fehlschlägt, dann wissen Sie, dass Ihr Programm noch einen Fehler aufweist und können es entsprechend verbessern bis alle Tests erfolgreich durchlaufen. *Achtung: Wenn alle Tests erfolgreich durchlaufen heisst dies nicht automatisch, dass ihr Programm komplett richtig ist. Die Tests können nur immer einen Teil aller möglichen Eingaben von ihrem Programm überprüfen.*

**Tests ausführen** Um die Tests automatisiert ausführen zu können, verwenden wir die Software *Gradle*. *Gradle* ist ein sogenanntes Buildtool, welches das kompilieren von Programmen sowie das Ausführen der Tests koordiniert.<sup>1</sup>

Um Ihre Programme zu testen öffnen Sie ein Terminal und wechseln in das Verzeichnis mit den von Ihnen heruntergeladenen Übungen. Dort finden Sie die Datei `gradlew`, mit der Sie *Gradle* starten können. Führen Sie folgendes Kommando aus:

```
> .\gradlew test (unter Windows)

> ./gradlew test (unter MacOS und Linux) 2
```

Dies testet jeweils alle Übungsaufgaben eines Übungsblattes und listet anschliessend auf, welche Prüfungen bestanden wurden und welche nicht. Für das erste Übungsblatt könnte die Ausgabe zum Beispiel wie folgt aussehen:

```
BMITests > testAufgabe2BerechnungRichtig PASSED
BMITests > testAufgabe2AusgabeRichtig PASSED
WelcomeTests > testAufgabe1EndetMitAusrufezeichen FAILED
    java.lang.AssertionError at WelcomeTests.java:40
WelcomeTests > testAufgabe1EnthaelthHalloUndName PASSED
WelcomeTests > testAufgabe1AufrufOk STANDARD_OUT
    Hallo Spock
WelcomeTests > testAufgabe1AufrufOk PASSED
6 tests completed, 1 failed
```

---

<sup>1</sup>Beachten Sie, dass Gradle nicht benötigt wird um Java zu programmieren. Es vereinfacht nur die Ausführung von wiederkehrenden Kommandos. Im Hintergrund werden aber einfach wieder die Standardkommandos `javac` und `java` aufgerufen. Für diese Vorlesung müssen Sie Gradle nicht verstehen, sondern lediglich die hier vorgegebenen Kommandos ausführen können.

<sup>2</sup>Falls Sie unter MacOS oder Linux die Fehlermeldung `-bash: ./gradlew: Permission denied` erhalten, dann müssen Sie die Datei `gradlew` zuerst mit dem Befehl `chmod +x gradlew` ausführbar machen. Danach sollte alles funktionieren.

Diese Ausgabe zeigt an, dass der Test `testAufgabe1EndetMitAusrufezeichen` fehlgeschlagen ist und somit Ihre Lösung für die entsprechende Teilaufgabe nicht richtig sein kann. In diesem Fall wurde das Ausrufezeichen am Ende der Ausgabe vergessen. Korrigieren Sie den Code und führen Sie die Tests nochmal aus.

**Lesen der Tests** Wenn Sie interessiert sind, wie ihr Code getestet wird, können Sie sich alle Testfälle anschauen. Sie finden den Code für die Testfälle im Verzeichnis `src/test/java`. Es handelt sich dabei um normalen Java Code. So ab Mitte der Vorlesung sollten Sie in der Lage sein den Testcode zu verstehen.