

Einführung in die Programmierung

Prof. Dr. Peter Thiemann
Marius Weidner, Hannes Saffrich
Lukas Kleinert, Timpe Hörig

Universität Freiburg
Institut für Informatik
Wintersemester 2023

Übungsblatt 1

Abgabe: Montag, 23.10.2023, 9:00 Uhr morgens

Wichtig

Bitte melden Sie sich bis spätestens Di, 17.10. über HisInOne für ein Tutorat an.

Aufgabe 1.1 (Setup)

Bevor es mit den eigentlichen Aufgaben losgeht, müssen Sie zunächst die für die Vorlesung benötigte Software installieren und sich mit unsere Webplattform zur Abgabe der Übungen vertraut machen. Schauen Sie sich hierzu die [Anleitung zur Installation](#) an. Die im Video gezeigte Zusammenfassung der Installationsschritte finden Sie [hier](#). Eine Erklärung zum Abgabesystem gibt es am Mi, 18.10 12:15-13:00 Uhr vor der Vorlesung (selber Raum), bzw. anschließend als Aufzeichnung auf der Vorlesungswebsite.

Sollten Sie bei der Installation Probleme haben, dann können Sie gerne im Chat¹ oder in den Tutoraten fragen. Wir sind auch ganz lieb und beißen nicht :)

Dateiformate und sonstige Regeln für Ihre Abgabe

- Verwenden Sie nur Befehle und Programmier Techniken, die Inhalt der *bisherigen* Vorlesungen (bis zum Abgabetermin) und Übungsblättern waren.
- Nach dem Hochladen ihrer Lösung führt der Buildserver style checks ihres Codes mit `Flake8` aus. Diese style checks müssen mit *success* beendet werden.
- Die einzelnen Aufgaben auf den Übungsblättern sind stets mit vollständigen Dateinamen annotiert. Bitte halten Sie sich an die Namen und die dazugehörigen Dateiformate. Beispielsweise soll Aufgabe 1.2 als Datei `algorithms.txt` abgegeben werden.
- Die Dateiformate sind immer in *plaintext* (UTF-8) und haben die Endungen `.txt` bzw. `.md` für Text und `.py` für Python-Code. Das heißt es sind insbesondere keine PDFs, keine Word-Dokumente und auch keine Bildschirmfotos erlaubt!
- Eine simple Möglichkeit diese Kriterien zu erfüllen ist es einfach alle Dateien mit Visual Studio Code zu erstellen bzw. zu bearbeiten. Wenn Sie zum Beispiel die Endung `.txt` verwenden, dann erkennt Visual Studio Code, dass es sich nicht um ein Python-Skript handelt und verhält sich wie ein normaler

¹Auf <https://git.laurel.informatik.uni-freiburg.de> mit Ihrem RZ-Account (wie bei Ilias) einloggen und oben im Menü auf "Chat" klicken.

Texteditor. Beim Bearbeiten von `.py` Dateien werden `Flake8`-Warnungen gelb markiert und mit einem Mouseover erklärt.

- Abgaben per Mail können nicht berücksichtigt werden. Geben Sie Ihre Aufgaben [über unser git-System](#) ab.

Aufgabe 1.2 (Algorithmen; 3 Punkte; Datei: `algorithms.txt`)

Entscheiden Sie, ob die folgende umgangssprachliche, prozedurale Beschreibung die Eigenschaften Präzision, Effektivität, statische Finitheit, dynamische Finitheit und Terminierung (siehe Folien) erfüllt. Begründen Sie jeweils kurz Ihre Antwort. Handelt es sich bei der Beschreibung um einen Algorithmus?

Gegeben seien zwei nicht-negative, ganze Zahlen x, y . Solange x ungleich y ist, führe den folgenden Schritt aus: Wenn x größer als y ist, ziehe y von x ab, ansonsten ziehe x von y ab. Schlussendlich, gib x aus.

Aufgabe 1.3 (Ausdrücke; 4 Punkte; Datei: `expressions.txt`)

In dieser Aufgabe geht es um einige Unterschiede zwischen dem “Skript-Modus” und dem “interaktiven Modus” von Python. Ein Python-Skript (Dateiendung `.py`) können Sie zum Beispiel ausführen, indem sie es in VS Code öffnen und rechts oben auf das `>`-Symbol klicken. Für den “interaktiven Modus” öffnen Sie ein Terminal und geben Sie den Befehl `python3.12` ein. Dass sich Python im “interaktiven Modus” befindet, erkennen Sie an den anführenden Prompt-Zeichen `>>>`. Beenden können Sie den Modus u.a. mit dem Befehl `exit()`.

- (a) (2 Punkte) Wenn Sie nacheinander die Ausdrücke `42` und `print(42)` in die interaktive Konsole eingeben, erhalten Sie die gleiche Ausgabe. Dennoch sind die beiden Ausdrücke nicht identisch. Versuchen Sie, Unterschiede herauszufinden, indem Sie die beiden Ausdrücke auch im “Skript-Modus” ausführen. Erklären Sie Ihre Erkenntnisse in eigenen Worten.
- (b) (2 Punkte) Wenn Sie `print((((3 * 3) + 13 // (5 - 1)) * (2 + 2))` in die interaktive Konsole eingeben, sehen Sie statt einer Zahl nur “...”. Warum? Was passiert, wenn Sie den gleichen Ausdruck als Skript ausführen?

Sie müssen für diese Aufgabe kein Python-Skript abgeben, sondern lediglich die Datei `expressions.txt` mit Ihren Erklärungen.

Aufgabe 1.4 (Print-Rätsel; 3 Punkte; Dateien: `print1.py`, `print2.py` und `print3.py`)

Ihr Python-Skript soll in den folgenden Teilaufgaben aus jeweils einer Zeile der Form `print(...)` bestehen, wobei Sie `...` durch einen Ausdruck ersetzen.

- (a) (1 Punkt) Schreiben Sie ein Python-Skript `print1.py`, das die folgende Ausgabe ² produziert:

²Ihre Ausgabe darf eine leere Zeile am Ende beinhalten

```
Python
Python
Python
```

Der String 'Python' darf dabei nur ein einziges Mal im `print`-Befehl vorkommen.

Folgende Python-Skripts sind also *keine* Lösung:

```
- print('Python\nPython\nPython')
- print('Python')
  print('Python')
  print('Python')
```

- (b) (1 Punkt) Schreiben Sie ein Python-Skript `print2.py`, das die folgende Ausgabe ² produziert:

```
HHH
HHH
HHH
```

Der String 'H' darf dabei nur ein einziges Mal im `print`-Befehl vorkommen.

- (c) (1 Punkt) Schreiben Sie ein Python-Skript `print3.py`, das die ersten 4 der letzten 4047 Ziffern der Zahl 4047^{4047} (im Dezimalsystem) berechnet und ausgibt. Angenommen, die letzten 4047 Ziffern der Zahl wären `27180470...`, dann soll die Ausgabe wie folgt aussehen:

```
2718
```

Aufgabe 1.5 (Erfahrungen; Datei: `NOTES.md`)

Notieren Sie Ihre Erfahrungen mit diesem Übungsblatt (benötigter Zeitaufwand, Probleme, Bezug zur Vorlesung, Interessantes, etc.).

Editieren Sie hierzu die Datei `NOTES.md` im Abgabeordner dieses Übungsblattes auf unserer Webplattform. Halten Sie sich an das dort vorgegebene Format, da wir den Zeitbedarf mit einem Python-Skript automatisch statistisch auswerten. Die Zeitangabe `3.5 h` steht dabei für 3 Stunden 30 Minuten.

Der Buildserver überprüft ebenfalls, ob Sie das Format korrekt angegeben haben. Prüfen Sie, ob der Buildserver mit Ihrer Abgabe zufrieden ist, so wie es im Video zur Lehrplattform gezeigt wurde.