

# Informatik I: Einführung in die Programmierung

Prof. Dr. Bernhard Nebel  
Tim Schulte, Thorsten Engesser  
Wintersemester 2017/2018

Universität Freiburg  
Institut für Informatik

## Übungsblatt 13

**Abgabe: Freitag, 02. Februar 2018, 20:00 Uhr**

**WICHTIGE HINWEISE:** Zur Bearbeitung der Übungsaufgaben legen Sie bitte ein neues Unterverzeichnis `sheet13` im Wurzelverzeichnis Ihrer Arbeitskopie des SVN-Repositories an. Ihre Lösungen werden dann in Dateien in diesem Unterverzeichnis erwartet. Beachten Sie bitte bei allen Aufgaben die *Hinweise zur Bearbeitung der Übungsaufgaben* unter der folgenden URL:

<http://gki.informatik.uni-freiburg.de/teaching/ws1718/info1/guide/hinweise.html>

Bewertet wird bei allen Aufgaben die letzte Version, die zur Deadline des Übungsblattes auf dem SVN-Server eingereicht ist.

**Aufgabe 13.1** (Regierungssysteme der Welt; Datei: `systems.py`; Punkte: 3+3+3)

In dieser Aufgabe geht es darum, aus den Wikipedia-Webseiten Informationen zu Staaten und deren Regierungssysteme zu extrahieren.

Die Liste der Staaten der Erde finden Sie unter der folgenden URL:

[https://de.wikipedia.org/wiki/Liste\\_der\\_Staaten\\_der\\_Erde](https://de.wikipedia.org/wiki/Liste_der_Staaten_der_Erde).

Die Liste der Regierungssysteme nach Staat finden Sie unter folgender URL:

[https://de.wikipedia.org/wiki/Liste\\_der\\_Regierungssysteme\\_nach\\_Staat](https://de.wikipedia.org/wiki/Liste_der_Regierungssysteme_nach_Staat).

- (a) Implementieren Sie eine Funktion `population()`, die ein Dictionary zurückgibt, welches jedem Staat die Gesamtzahl seiner Einwohner zuordnet.
- (b) Implementieren Sie eine Funktion `government()`, die ein Dictionary zurückgibt, welches jedem Staat die Bezeichnung seines Regierungssystems zuordnet.
- (c) Schreiben Sie ein Programm, welches für jedes Regierungssystem die folgenden Informationen ausgibt:
  - Den Namen des Regierungssystems,
  - die Gesamtzahl der Einwohner aller zugehörigen Staaten,
  - deren prozentualer Anteil an der Weltbevölkerung, sowie
  - die jeweiligen Staaten als komma-separierte Liste, absteigend sortiert nach Einwohnerzahl.

Verwenden Sie hierzu die in (a) und (b) definierten Funktionen. Beachten Sie für die Zusammenführung der Datensätze, dass der gleiche Staat auf den beiden Wikipedia-Seiten unterschiedliche Bezeichnungen haben kann (z.B. Palästina und Palästinensische Autonomiegebiete). Sie dürfen diese Ausnahmen im Code manuell behandeln. Achten Sie bei der Ausgabe auf eine ansehnliche Formatierung.

*Hinweis:* Benutzen Sie bei der Bearbeitung dieser Aufgabe die in der Vorlesung vorgestellten Techniken zum Scraping von Webseiten sowie reguläre Ausdrücke (keine zusätzlichen Python-Libraries).

**Aufgabe 13.2** (Tupel vs. Listen; Punkte: 3+3+3, Dateien: `test_tuple_list.py`)

*Tuples are faster than lists.* Diese Feststellung findet sich in einigen Online-Dokumentationen zu Python, aber stimmt das? Um diese Frage zu analysieren, interessiert uns:

- (a) Ist der Zugriff auf einen Listeneintrag `x = lst[k]` langsamer als der Zugriff auf einen Tupeleintrag `x = tpl[k]`?
- (b) Ist die Erzeugung eines Listen-Slices `x = lst[k:n]` langsamer als die Erzeugung eines Tupel-Slices `x = tpl[k:n]`?
- (c) Ist die Erzeugung einer Liste durch `lst = [x1, x2, ..., xN]` mittels Variablen `x1, x2, ..., xN` (den Variablen müssen natürlich vorher Werte zugewiesen sein) langsamer als die Erzeugung eines entsprechenden Tupel `tpl = (x1, x2, ..., xN)`?

Diskutieren Sie obige Fragestellung anhand eigener Testergebnisse (als Kommentar in der Datei `test_tuple_list.py`). Verwenden Sie dazu das `timeit`-Modul und eines der beiden auf der Webseite der Übungen bereitgestellten Templates. Gestalten Sie die Ausgabe so, dass ersichtlich wird, für welche der Fragen (a) oder (b) oder (c) und welchen Datentyp Ihre Zeitmessungen gelten.

**Aufgabe 13.3** (Erfahrungen; Datei: `erfahrungen.txt`; Punkte: 2)

Legen Sie im Unterverzeichnis `sheet13` eine Textdatei `erfahrungen.txt` an. Notieren Sie in dieser Datei kurz Ihre Erfahrungen beim Bearbeiten der Übungsaufgaben (Probleme, benötigter Zeitaufwand nach Teilaufgabe, Bezug zur Vorlesung, Interessantes, etc.).