

# Einführung in die Programmierung mit Skriptsprachen

## Aufgabenblatt 8

Abgabe: 10.01.2023 18:30

Abnahme: 11.01.2023

---

### 1. Spickzettel

5 Pkt.

Erstellen Sie für den seit dem E-Assessment 1 behandelten Stoff einen neuen Spickzettel von Hand, einseitig beschrieben.

### 2. Gültigkeitsbereiche

Schreiben Sie ein Programm in Python, das Namensräume benutzt. Neben dem Hauptprogramm soll es die Funktionen *scopes*, *do\_local*, *do\_nonlocal* und *do\_global* geben, die alle keinen Wert zurückgeben. Es soll in den Funktionen und im Hauptprogramm jeweils nur eine Variable namens *wort* (frei oder gebunden) vorkommen.

Am Anfang jeder Funktion und des Hauptprogramms gibt es eine Zuweisung an *wort*, keine weiteren. Jede Funktion, sowie das Hauptprogramm, hat ein Print-Statement, das den Namen der Funktion (also von sich selbst) und den Inhalt von *wort* ausgibt. Nach einem Funktionsaufruf wird vom aufrufenden Programmteil wie oben wieder der Name des Aufrufers und der Inhalt von *wort* ausgegeben.

Die Ausgabe sieht folgendermaßen aus:

```
im Hauptprogramm: test wort
in scopes: scopes wort
in do-local: local wort
in scopes: scopes wort
in do-nonlocal: nonlocal wort
in scopes: nonlocal wort
in do-global: global wort
in scopes: nonlocal wort
im Hauptprogramm: global wort
```

Ermitteln Sie, wie die Funktionen ineinander geschachtelt definiert und aufgerufen werden müssen, und auf welche Variable sich jeweils bezogen werden muss, damit diese Ausgabe entsteht.

### 3. Parameterübergabe in Python

Entwickeln Sie ein Python-Programm, mit dem man herausfinden kann, ob ein Parameter mit dem call-by-value-Mechanismus, mit call-by-reference oder evtl. ganz anders weitergegeben wird. Testen Sie dabei Parameter von unterschiedlichen Datentypen: Integer, String, Listen.

Schreiben Sie dazu ein kurzes Programm mit Funktionen, das aktuelle Parameter von den oben genannten mutable und immutable Typen benutzt, deren Werte in den Funktionen geändert werden. Für Listen schreiben Sie eine Funktion, die in einer übergebenen Liste ein Element ändert, und eine Funktion, die der als Parameter übergebenen Liste eine komplett andere Liste zuweist. Alle Funktionen sollen nichts zurückgeben.

Geben Sie die Werte der aktuellen Parameter im aufrufenden Programm vor und nach den Funktionsaufrufen aus.

Skizzieren Sie den Ablauf im Stack bei Funktionsaufrufen in Python mit den Parametern. Können Sie hier call-by-reference oder call-by-value identifizieren?