

Übung 1

Abgabe 17.4.2008

Aufgabe 1 – Einführung in Python

10 Punkte

- Installieren Sie die Programmiersprache Python auf Ihrem System (siehe www.python.org). Wenn Sie Linux verwenden, ist Python wahrscheinlich bereits installiert.
- Arbeiten Sie aus dem Python-Tutorium (siehe docs.python.org/tut/tut.html) die Abschnitte 1 bis 7 durch. Sie sollen dabei die angegebenen Beispiele auf Ihrer Maschine nachvollziehen. Die folgenden Unterabschnitte (die sich an fortgeschrittene Programmierer richten) können übersprungen werden: 2.2, 4.7, 5.1.3, 5.1.4, 5.6, 5.7, 5.8, 6.4.
- Implementieren Sie das Sieb des Eratosthenes zur Bestimmung von Primzahlen (den Algorithmus finden Sie im Internet). Geben Sie Ihre Lösung in einem File „sieve.py“ ab, das wie folgt benutzt werden kann:

```
>>> execfile("sieve.py") # load the function sieve()
>>> primes = sieve(1000) # return an array of all primes below 1000
```

Aufgabe 2 – Werte und Referenzen

10 Punkte

- Wenn Sie im Internet auf eine interessante Seite stoßen, die Sie sich merken wollen, können Sie entweder ein Lesezeichen setzen (bookmark), oder die Seite abspeichern (save page as). Dies entspricht genau der Unterscheidung zwischen Referenzen und Werten in der Programmierung. Beschreiben Sie Vor- und Nachteile beider Vorgehensweisen. Denken Sie dabei z.B. an Speicherverbrauch, Offline-Betrieb, Datenkonsistenz und die Möglichkeit, dass die Originalseite aktualisiert oder gelöscht werden könnte.
- In Python erfolgen Zuweisungen standardmäßig per Referenz.¹ Für alle Datentypen kann man eine Zuweisung per Wert erzwingen, wenn man das Python-Modul "copy" verwendet. Informieren Sie sich über dieses Modul (siehe docs.python.org/lib/module-copy.html). Analog gilt für Vergleiche, dass der operator "==" (z.B. a == b) die Werte von zwei Variablen vergleicht, während der operator "is" (z.B. a is b) prüft, ob zwei Variablen dasselbe Objekt referenzieren. Führen Sie folgendes Experiment durch:

```
>>> import copy
>>>
>>> a = [1, 2, 3, 4]
>>> b = a
>>> c = copy.deepcopy(a)

>>> print a == b, a == c, b == c # are they equal?
>>> print a is b, a is c, b is c # are they identical?

>>> a[0] = 42 # change array a
>>> print a == b, a == c, b == c # are they still equal?
>>> print a is b, a is c, b is c # are they identical?
>>> print a[0], b[0], c[0] # which values do you get?
```

Beschreiben und begründen Sie, wie sich das Programm verhält.

¹ Ausnahmen bilden die Zahlen (Typen bool, int und float), die stets als Werte zugewiesen werden.