Institut für Informatik
Dr. Martin Buchholz
Benjamin Peherstorfer

Technische Universität München
WS 2010/2011

Einführung in die wissenschaftliche Programmierung

Übungsblatt 5

1.) Telefonbuch

Erweitern Sie das Telefonbuch von Blatt 4 so, dass fehlerhafte Eingaben erkannt werden. Z.B. darf ein Name keine Zahl beinhalten, oder eine Nummer keinen Buchstaben, usw.

Benutzen Sie reguläre Ausdrücke um falsche Eingaben zu erkennen.

2.) Endlicher Automat

Schreiben Sie eine Funktion, welche `True` zurück gibt, falls der übergebene String `s` den Ausdruck `re.match('^a(b{3}|ab)*$', s)` erfüllt. Benutzen Sie nicht die eingebaute Reguläre-Ausdrücke-Engine von Python (d.h. das Modul `re`).

3.) E-Mail Crawler

In dieser Aufgabe soll ein Programm zum Auslesen von E-Mail Adressen auf Webseiten entwickelt werden. Das Programm liest eine Webseite ein und extrahiert sämtliche Links und E-Mail Adressen auf der Webseite. Die E-Mail Adressen werden gespeichert (und ausgegeben). Dann wird einem Link von der Webseite gefolgt um dort ebenfalls Links und E-Mail-Adressen zu extrahiert.

Mit dem Python Modul `urllib` können sehr einfach Webseiten als String ausgelesen werden. Dazu müssen Sie nur

```
site = urllib.urlopen('http://www.google.com').read()
```

aufrufen. Der Quelltext der Webseite steht dann in der Variable `site`.

E-Mail Crawler

Implementieren Sie nun einen E-Mail Crawler:

i) Schreiben Sie eine Funktion, welche aus einem String E-Mail Adressen extrahiert. Im (HTML-)Quelltext einer Webseite treten E-Mail Adressen entweder einfach als Wort (durch Leerzeichen begrenzt) auf, oder aber in der Form

```
<a href="mailto:example@test.com">Herr Example</a>
```

Benutzen Sie reguläre Ausdrücke und die Funktion `re.finditer`.

ii) Schreiben Sie eine Funktion, welche Webadressen (z.B. http://de.wikipedia.com, http://www.tum.de/, usw.) aus einem String extrahiert. Wieder treten Adressen in zwei Formen auf. Entweder einfach durch Leerzeichen begrenzt, oder in der Form

```
<a href="http://www.facebook.com">Facebook</a>
```

iii) Starten Sie mit einer beliebigen Webseite und lesen Sie den Quelltext mit der `urllib` Bibliothek ein. Extrahieren Sie dann die E-Mail Adressen und speichern Sie diese in einer Menge (`set`). Extrahieren Sie die Links und speichern Sie diese in einer Liste. Entnehmen Sie dann ein Element der Liste mit Links und starten Sie den Vorgang erneut.

Fehlerbehandlung (Exceptions)

Des Öfteren werden Sie auf ungültige Links treffen. Die `urlopen` Funktion von `urllib` wirft in so einem Fall eine `IOError` Exception. Fangen Sie diese ab und reagieren Sie geeignet darauf (z.B. überspringen der ungültigen Adresse).

Das Programm läuft so lange bis die Liste mit Links leer ist. Oftmals möchte man das Programm aber nach einiger Zeit mit Strg-C einfach abbrechen und dann die gefundenen E-Mail Adressen ausgeben. Python wirft in diesem Fall eine `KeyboardInterrupt` Exception. Fangen Sie auch diese an geeigneter Stelle ab, geben Sie dann die gefundenen E-Mail Adressen aus und beenden Sie das Programm.