



Übungsblatt 9

Willkommen zum Praktikum zu Programmieren 3. Lesen Sie die Hinweise.

Aufgabe 1. In dem Bild `mi.png` ist eine Nachricht versteckt. Welche Nachricht ist es? Schreiben Sie die Nachricht "Nein, gar nicht. Es war ganz leicht." statt dessen in das Bild und speichern Sie es ab. Verwenden Sie die Python Image Library (PIL). Importieren Sie dazu das Modul `Image`. Mit `Image.open` öffnen Sie ein Bild. Finden Sie weitere Möglichkeiten von `Image` selbst heraus. Schreiben Sie ein Skript `message.py`, das Sie von der Kommandozeile aufrufen können mit

```
$ message.py -w "Die Nachricht" mi.png
```

zum schreiben der Nachricht und

```
$ message.py mi.png  
Die Nachricht
```

zum Lesen der Nachricht. Warum verwenden wir hier PNG und nicht JPG?

Aufgabe 2. Schreiben Sie ein Programm `filter.py`, das mit zwei Dateinamen als Parameter – `<lesebildname>`, ein Dateiname eines zu lesenden Bilds b_r als ersten Parameter und `<schreibbildname>`, ein Dateiname eines zu schreibenden Bilds b_w als zweiten Parameter – aufgerufen wird. Das eingelesene Bild b_r wird nach einer "Binarisierung" zu b_w und wieder abgespeichert. Bei der Binarisierung werden alle Pixel, die über einem gewissen Helligkeits-Schwellwert liegen weiß und alle anderen schwarz. Sie können davon ausgehen, dass Sie Farbbilder in RGB-Darstellung haben. Der aktuellen Helligkeitswert h ($0 \leq h \leq 255$) ist die Summe der Werte der drei Farben durch drei. Erlauben Sie noch zusätzlich einen optionalen Kommandozeilenparameter `-f <wert>` mit einer Angabe des Helligkeitsschwellwerts in ganzzahligen Prozent zwischen 0 und 100.

- Realisieren Sie den Filter zunächst in Python mit ähnlichen Hilfsmitteln, wie in Aufgabe 1.
- Da diese Aufgabe eine typische Aufgabe für maschinennahes Programmieren ist schreiben Sie ein C-Modul, das die eigentlich Filterung für Sie übernimmt. Verwenden Sie `ctypes` zur Integration. Die Methoden `tostring` und `fromstring` werden helfen. Ist die C-Variante schneller?

Aufgabe 3. Schreiben Sie reguläre Ausdrücke zum Matchen folgender Objekte.

- Postleitzahlen in Deutschland
- Datum in der Form TT.MM.JJJJ, (Tag.Monat.Jahr)



- Euro-Beträge: Optional mit durch Leerzeichen getrenntem EUR. Optional zwei durch Komma getrennte Nachkommastellen. Der Punkt ist erlaubt als Separator für Tausenderbeträge.
- Telefonnummern: Optional mit Länderkennung (z.B. +49) durch Leerzeichen abgesetzt und ohne 0 bei der Vorwahl. Als Trennzeichen sind Leerzeichen, / und - erlaubt. Zwischen zwei Trennzeichen sind mindestens 2 Ziffern notwendig.
- Email-Adressen: Vor dem @ beliebige Buchstaben- und Ziffern-Kombinationen mit führendem Buchstaben. Optional getrennt durch ein Punkt oder ein -. Muss mit Buchstaben oder Ziffer enden. Hinter dem @ alles in Kleinschreibung und mindestens ein Domainname plus Länderkennung.

Testen Sie Ihre regulären Ausdrücke in Python. Nutzen Sie dazu die Datei `reg.txt`. Schauen Sie sich die Struktur der Datei an und entwickeln Sie ein passendes Skript um Ihre Ausdrücke zu testen. Welche Zeichenketten richtig sind ist als "dritte" Spalte angegeben. Diese Spalte muss nicht verarbeitet werden.

Aufgabe 4. Sie sind Web-Master und wollen Ihre Log-Dateien analysieren. Sie analysieren die Dateien `/var/log/apache2/access_log*` auf dem Rechner `www` (die aktuellen Log-Dateien des Web-Servers unseres Studiengangs). Lesen Sie das Format der Log-Datei nach unter

<http://httpd.apache.org/docs-2.2/logs.html>

Sie möchten folgendes wissen: Wie häufig wurden Bilder (jpg, gif, png) geladen? Wie viele unterschiedliche Bilder wurden je Typ geladen? Welche Zugriffe auf Web-Seiten resultierten aus Suchanfragen von Google und wie lauteten diese Suchanfragen? Schreiben Sie ein ausführbares Python-Skript, das Ihnen diese Fragen beantwortet.

Hinweis 1. Sie erzeugen eine "shared Library" durch Angabe von `-fPIC -shared` als Kommandozeilenoption für `gcc`. Das Ergebnis sollte die Endung `.so` haben und kann mit Hilfe von `ctypes` mit der Funktion `cdll.LoadLibrary` geladen werden. Beachten Sie, dass Sie den aktuellen Pfad zum Laden noch mit zum Dateinamen dazu nehmen müssen.

<http://www.mi.hs-rm.de/~barth/hsrm/prog3>