Übungen zu Computergrundlagen WS 2015/2016

Übungsblatt 3: Shellskripte

06.11.2015

Allgemeine Hinweise

- Abgabetermin für die Lösungen ist
  - **Freitag, 13.11., 11:00**
- Schickt die Lösungen bitte per Email an Euren Tutor.

Aufgabe 3.1: Bilder per Email (5 Punkte)

Betrachte das folgende Skript (`/group/cgl/2015/03/doit.sh`):

```
#!/bin/bash
#
# A useful script - if it wasn't so buggy...
#

# set variables
defaultdir=.
maxsize= 640x640

# command line handling
dir=$1
if [ -z $dir ] ; then
    dir=$defaultdir
fi

# find images
images=$(ls $dir/*.jpg)

if [ -z "$images" ];
    echo "No images in $dir!"
    exit 2
fi

# resize images
for image in $images do
    dirname=$(dirname image)
    basename=$(basename $image .jpg)
    newimage=$dirname/${basename}_small.jpg
    echo "$image -> $newimage"
    convert $image -resize max_size $newimage
fi
```

Als Argument liest das Skript einen Verzeichnisnamen. In diesem Verzeichnis sollten sich einige `.jpg`-Bilder befinden.

- **3.1.1** Welche Aufgabe soll das Shellskript vermutlich erfüllen? Wofür ist es nützlich? (1 Punkt)

- **3.1.2** Kopiert Euch das Shellskript in Euer Heimatverzeichnis. Was müsst Ihr machen, um diese Datei durch `./doit.sh` aufrufen zu können? (1 Punkt)
- **3.1.3** Ruft das Skript wie folgt auf:

```
./doit.sh /group/cgl/2015/03/images
```

  Wenn Ihr diesen Befehl ausführt, gibt es verschiedene Fehler. Dabei gibt es zwei Arten von Fehlern: einerseits Fehler im Skript, die seine Ausführung verhindern, andererseits wird das Skript aber auch nicht so verwendet, wie der Autor sich dies wohl gedacht hat.

  Korrigiert die Fehler im Skript und gebt das korrigierte Skript als Lösung ab! (2 Punkte)

  Hinweise:

  – Nicht immer geben die Fehlermeldungen einen direkten Hinweis darauf, was schief gelaufen ist. Alle Fehler in diesem Skript können durch das Hinzufügen oder Verschieben von wenigen Zeichen korrigiert werden.

  – Ein erstaunlich nützliches Werkzeug zur Fehlersuche in solchen Skripten ist ein guter Editor, der die unterschiedlichen Elemente eines Shellskripts unterschiedlich einfärbt („Syntax-Highlighting“). Schaut Euch genau an, wie der Editor die verschiedenen Elemente einfärbt, und welche davon nicht wie erwartet aussehen.

- **3.1.4** Auf dem gegebenen Verzeichnis `/group/cgl/2015/03/images` kann das Skript so nicht funktionieren. Warum? (1 Punkt)

Aufgabe 3.2: Webmaster (5 Punkte)

Ein typischer Webserver erzeugt Logdateien, die jeden Zugriff durch Zeilen der Form

```
127.0.0.1 - - [24/Mar/2011:16:56:12 -100] ‘‘GET http://www.google.de HTTP/1.1’’ 200 2459
IP  - - [Timestamp Zone] ‘‘Command Webpage Format’’ Returncode size
```

dokumentieren. Hieraus lässt sich sehr gut analysieren, welche Seite wie oft von wem aufgerufen wurde.

Schreibt ein Skript `analyze_log.sh`, das als einzigen Parameter den Namen einer solchen Logdatei bekommt. Danach gibt es zu dieser Logdatei folgendes aus:

- Wie viele *verschiedene* Webseiten angefragt wurden. Die Webseite steht immer in der 7. Spalte (jedes Leerzeichen trennt Spalten voneinander).
- Wie oft jede Webseite aufgerufen wurde. Dazu sucht Euch zunächst heraus, welche Webseiten es überhaupt gibt. Diese stehen immer in der siebten Spalte; an dem „http://“ stören wir uns diesmal nicht. Anschließend schreibt eine Schleife, die für jede Webseite die Anzahl der Aufrufe ausgibt.

Hinweise:

- Wenn Ihr in Schritten vorgeht, ist es einfacher.
- Hilfreiche Werkzeuge für diese Aufgabe sind `sort`, `wc`,`grep`, `awk` (und `man`)
- Mithilfe von `awk` können zum Beispiel einzelne Spalten aus einer Datei gelesen werden. Der folgende Programmausschnitt gibt die 2. Spalte aus der Datei “Textdatei” wieder.

```
awk '{print $2}' Textdatei
```