

Übungen zu Computergrundlagen WS 2019/2020

Übungsblatt 3: Shellskripte

8. November 2019

Allgemeine Hinweise

- Abgabetermin für die Lösungen ist **Freitag, 15.11.2019, 11:00 Uhr**
- Schickt die Lösungen bitte per Email an Euren Tutor:
 - Montag 14:00–15:30: Moritz Schumacher (mschumacher@icp.uni-stuttgart.de)
 - Dienstag 9:45–11:15: Samuel Tovey (stovey@icp.uni-stuttgart.de)
 - Dienstag 15:45–17:15: Philipp Stärk (pstaerk@icp.uni-stuttgart.de)
 - Mittwoch 15:45–17:15: Marco Brückner (mbrueckner@icp.uni-stuttgart.de)
 - Donnerstag 9:45–11:15: Ingo Tischler (itischler@icp.uni-stuttgart.de)
- Die Übungen sollen von Gruppen von jeweils *zwei* (nur in Ausnahmefällen drei) Leuten bearbeitet werden. Bitte gebt *nur eine Lösung pro Gruppe* ab und nennt in eurer Abgabe alle Mitglieder eurer Gruppe!
- Wie in den vorherigen Übungsblättern sollen die Lösungen der Aufgaben in eine Textdatei eingetragen werden, welche ihr dann per E-Mail an euren Tutor schickt.

Aufgabe 3.1: Bilder per Email (5 Punkte)

Betrachte das folgende Skript (/group/cgl/2019/03/doi.sh):

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

# set variables
defaultdir=.
max_size= 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 ausführen zu können? (1 Punkt)
- **3.1.3** Ruft das Skript wie folgt auf:

```
./doit.sh /group/cgl/2019/03/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/2019/03/images kann das Skript so nicht funktionieren. Warum? (1 Punkt)

Aufgabe 3.2: Backup-Skript (5 Punkte)

Erstelle ein Skript, dass alle *wichtigen* Dateien in deinem Home-Verzeichnis sichert. Benutze den Befehl tar um ein gzip-komprimiertes Archiv zu erstellen.

- Die Backup-Dateien werden nach dem Zeitpunkt des Backups in der Form „backup-JJJJ-MM-TT-HH-MM.tar.gz“ benannt und in ein eigenes Unterverzeichnis ~/backup gespeichert. Ein am 2. 11. um 13:00 Uhr erzeugtes Backup soll also „backup-2019-11-02-13-00.tar.gz“ heißen. (2 Punkte)
- Existiert das Verzeichnis ~/backup noch nicht, soll es erzeugt werden. Existiert es, ist aber eine Datei, soll es eine Fehlermeldung geben. (1 Punkt)

- Das Skript soll stets das gesamte Home sichern, egal, von wo aus es aufgerufen wird. Ausnahmen sind die Verzeichnisse ~/backup (klar, dort liegen ja die Backups) ~/.local und ~/.cache (diese können sehr groß werden). Diese Verzeichnisse bitte auslassen. (1 Punkt)
- Das Skript soll den Benutzer genau über alles informieren, was es tut. (1 Punkt)

Hinweise:

- Das Datum bekommt man mit Hilfe von `date`. Um die Ausgabe des Befehls als Zeichenkette in eine Variable `VAR` zu schreiben, benutze `VAR=$(date)`.
- Der Übungsleiter kann Fehler nur korrigieren, wenn er verstehen kann, was Deine Intention war. Daher bitte reichlich kommentieren. Ein Kommentar sollte *nicht* beschreiben, was die Befehle tun, sondern, was der Gedanke dahinter ist!
- Beim Testen bitte auf den verfügbaren Speicherplatz (300MB) achten. `du -sh ~` sagt Dir jederzeit, wieviel Platz Du belegst. Eventuell musst Du dann Backups löschen.
- Wenn Du in Schritten vorgehst, ist es einfacher. Probiere zunächst auf dem Terminal, wie Du den Dateinamen des Backups erzeugen kannst, und benutze erst dann `tar`.