

Aufgabe 3.1

Dateien umleiten und erstellen:

- a) `ls -aR > ls.txt` : Ausgabe mit `>` in eine Datei `ls.txt` umleiten
- b) `cat meineErsteDatei >> ls.txt` : Anfügen von Inhalt in eine existierende Datei
- c) `cat > yes` ,dann Return dann „y“ und dann Return und Strg+D : Datei `yes` mit Inhalt `y` erstellen
`cat > no` ,dann Return dann „n“ und dann Return und Strg+D
- d) `cat < no : ? listet Inhalt von no auf`
`cat < yes`
- e) `ls -aR | more` : Erster Befehl wird weitergeleitet, Datei nicht im zweiten Befehl erwähnen
- f) `rm -i ls.txt < no` : es wird abgefragt ob die Datei `ls.txt` wirklich gelöscht werden soll

Aufgabe 3.2

Datei `.gz` entpacken:

```
tar xzvf uebung3.tar.gz
```

Aufgabe 3.3

Anzeigen von bestimmten Menge von Zeilen einer Datei:

- a) `cd uebung3/Aufgabe3.3` `cat test` Inhalt ausgeben: Zeilen 1 bis 50
- b) `head -8 test` : Nur die ersten acht Zeilen der Datei werden ausgeben
- c) `tail -10 test` : Nur die letzten zehn Zeilen der Datei werden ausgeben
- d) `sed -ne '11,15p' test` : Nur Zeile elf bis 15 der Datei werden ausgeben

Aufgabe 3.4

Um welche Dateitypen handelt sich es:

`0_Vorspann.pdf` ist ein PDF-Dokument

`Test` ist ein ASCII text

`Passwd` ist ein ASCII text

`Ls` ist ein ELF 64-bit LSB shared objekt

Der Befehl lautet : `file dateiname` oder `file /verzeichnis/dateiname`

Aufgabe 3.5

Wörter/Zeilen/Zeichen mit `wc` (word count) ausgeben

- a) `wc -w Kafka` -> 2546 Wörter : gibt die Anzahl der Wörter aus
- b) `wc -l Kafka` -> 280 Zeilen : gibt die Anzahl der Zeilen aus
- c) `wc -m Kafka` -> 16134 Zeichen : gibt die Anzahl der Zeichen aus
- d) `wc -c Kafka` -> 16134 Länge (in Bytes) : gibt die Länge in Bytes aus

Aufgabe 3.6

`grep` = sucht nach bestimmten eingegeben Muster/Parametern

- a) `grep -w Waechter Kafka` Zeigt alle Zeilen mit Waechter an, Waechter wird **gehighlightet**
`grep -w Waechter Kafka >> Aufgabe3.5.Waechter` : erzeugt Datei, die alle Zeilen enthält in den `Waechter` **vorkommt**
- b) `grep -w Waechter Kafka | wc -l` -> es sind 17 Zeilen : gibt die Anzahl der Zeilen aus, in denen `Waechter` **vorkommt**
- c) `grep -v Waechter Kafka >> Aufgabe3.5.NichtWaechter` : erzeugt eine Datei mit allen Zeilen, in denen `Waechter` **nicht vorkommt!**

Aufgabe 3.7

Dateien sortieren und in neue Datei kopieren

- a) `sort a >> a.sorted` : Inhalt der Datei wird sortiert, hier alphabetisch, neue Datei `a.sorted` mit dem sortierten Inhalt wird erzeugt
- b) `sort -n b >> b.sorted` : Inhalt der Datei wird sortiert, hier von klein nach groß (Zahlen), neue Datei `b.sorted` wird mit dem sortierten Inhalt erzeugt
- c) `sort -n -u b >> c.sorted` : Inhalt der Datei wird von klein nach groß sortiert, doppelte Zahlen werden entfernt und dann der Inhalt in `c.sorted` geschrieben

Aufgabe 3.8 muss nicht gelöst werden

Aufgabe 3.9

Dateityp anzeigen und Datei öffnen

- a) `file ref-emacs.ps` -> PostScript Dokument : Dateityp herausfinden
- b) `gv ref-emacs.ps` : PostScript Dokument öffnen

Aufgabe 3.10

- a) Verzeichnis betrachten:

```
ls -l :listet alle Dateien & dazugehörigen Infos auf
dfunk001@scooter:~/einf-Inf/uebung3/Aufgabe3.10$ ls -l
total 8
-rw-r--r-- 2 dfunk001 students 59 Nov  9 2016 text1
lrwxrwxrwx 1 dfunk001 students  5 Oct 29 2010 text2 -> text1
-rw-r--r-- 2 dfunk001 students 59 Nov  9 2016 text3
```

- b) Inhalt der Dateien auflisten:

```
cat text1 bzw. text2, text3
```

In den Dateien `text1`, `text2` und `text3` steht vollgendes:

Das ist ein kleiner Text.

Aendere mich doch mal ein wenig!

Nach ändern von `text1` mit dem Befehl `nano`:

Das ist ein kleiner Text.

Aendere mich doch mal ein wenig!

Ich habe jetzt nur etwas in die Datei `text1` geschrieben!

Der Inhalt aller drei Dateien ändert sich, obwohl nur `text1` geöffnet und dort hinein geschrieben wurde!

Die Datei `text2` ist ein „symbolic link“ auf die Datei `text1`

- c) OFFEN
- d) OFFEN

Aufgabe 3.11

Belegten Speicher und max Speicher anzeigen lassen:

- a) `quota` -> es sind 41764 Bytes auf dem Server belegt
- b) auch mit `quota` -> es dürfen maximal 10.895.369 Bytes belegt werden (in etwa 10GB)

Aufgabe 3.12

Verzeichnis archivieren & .gz erstellen, USB-Stick mounten, Dateien auf USB kopieren:

- a) `tar -c -f myuebung3.tar uebung3` : erstellt das tar-Archiv `myuebung3.tar`
- b) `gzip myuebung3.tar` : erstellt aus `.tar` eine `.tar.gz` Datei
- c) `mkdir USB` : Mountverzeichnis erstellen
 - `sudo mount /dev/sda1 home/pi/USB` : USB-Stick mit Mountverzeichnis verknüpfen
 - `cp Datei1 USB/` : Datei auf USB-Stick kopieren
 - `sudo umount /dev/sda1` : USB-Stick auswerfen