

## Aufgabe 1 – *Spickzettel*

Mit dem Werkzeug `wget` kannst du Dateien aus dem Internet herunterladen, ohne dass du eine grafische Oberfläche benötigst.

- Lade mit `wget` die Datei `http://fsi/vorkurs/tippst_du_noch_oder_tabst_du_schon.tex` in dein *Home*-Verzeichnis (`wget <URL>`).
- Erstelle nun ein Unterverzeichnis `Vorkurs/cheatsheet/` in deinem *Home*-Verzeichnis, verschiebe die Datei dorthin und benenne sie in `cheatsheet.tex` um.

### — Exkurs: $\LaTeX$ ... aber nicht zum Anziehen —

Der Spickzettel wurde mit der Sprache  $\LaTeX$  erstellt. Schau dir die Datei `cheatsheet.tex` mit einem Editor deiner Wahl (z. B. `kwrite`) an. Was fällt dir auf?

Im normalen Text sind spezielle Befehle eingebaut, die das Aussehen der Ausgabeseite beeinflussen. Ersetze in Zeile 22 „Hier könnte dein Name stehen“ durch deinen Vor- und Nachnamen, um deinen Spickzettel zu personalisieren.

Wenn du den Spickzettel erweitern willst, verwende folgende Syntax:

```
\cmdbox{überschrift}{
  \cmd{befehl}{argumente}{beschreibung}
  \cmd{befehl}{argumente}{beschreibung}
}
```

`\cmd` kannst du beliebig oft untereinander schreiben.

Sollten wir dein Interesse an dieser mächtigen Setzsprache geweckt haben, so solltest du dich hier (<http://www.uni-giessen.de/hrz/tex/cookbook/cookbook.html>) mal einlesen.

### — Ende Exkurs —

- Prüfe mit `ls`, ob die Datei `cheatsheet.tex` im **aktuellen** Verzeichnis existiert, und erstelle dann mit dem Befehl `pdflatex cheatsheet.tex` ein PDF aus der Textdatei.
- Betrachte die entstandene Datei `cheatsheet.pdf` mit einem PDF-Viewer (z. B. `kpdf`).

## Aufgabe 2 – *Drucken im CIP*

Der Befehl `lpr` fügt einen Druckauftrag in die Warteschlange des Druckers ein (und druckt diesen dann). Mit dem Kommando `lpq` erfährst du, an welcher Position in der Warteschlange des Druckers dein Druckauftrag steht. In den ersten beiden Stockwerken gibt es mindestens zwei Drucker (a, b und evtl. c) in den großen CIP-Pools. Deren Beschriftung setzt sich wie folgt zusammen:

$$ps\langle\text{Nummer Des Stockwerks}\rangle\langle\text{Buchstabe Des Druckers}\rangle\text{cip}$$

- Überprüfe erst mal mit `pr_acct` dein Druckerkonto. Du solltest über ein gewisses Startguthaben verfügen<sup>1</sup>. Ist dein Startguthaben aufgebraucht, kannst du dein Druckkonto im Sekretariat (gegenüber vom CIP 2) aufladen lassen.

---

<sup>1</sup>Wenn nicht, wende dich an einen Tutor.

- Schau nach, welcher Drucker in deinem Stockwerk die kürzeste Warteschlange hat (`lpq -P<Druckername>`). Überprüfe nur die Druckernamen, die auf `d` enden. Solltest du den Namen der Drucker gerade nicht parat haben, schau einfach auf dem Drucker nach.
- Drucke die Datei `CheatSheet.pdf` mit `lpr -P<Druckername> <Dateiname>` auf einem Drucker **deines** Stockwerkes aus.
- Sobald dein Ausdruck fertig ist, hole dir diesen vom Drucker ab. Den Spickzettel könntest du im Verlauf der Kurse vielleicht noch einmal brauchen.

## Aufgabe 3 – Quota

In deinem *Home*-Verzeichnis hast du anfangs 150 MB Speicherplatz zur Verfügung – und der ist überraschend schnell voll. Um zu überprüfen, wie viel Platz noch frei ist, kann man (natürlich in einer Shell) den `quota`-Befehl verwenden:

- `blocks`: der belegte Speicher
- `quota`: der zur Verfügung stehende Speicher

Schau in der Manpage von `quota` nach, wie man die Ausgabe auf „human readable“ stellen kann!

Falls du herausgefunden hast, dass du zu viel Speicher belegst, kannst du mit dem Programm `du` (für *disk usage*) nachschauen, welche Verzeichnisse wieviel Speicherplatz belegen. Schau nach, welches Verzeichnis in deinem Home den meisten Speicherplatz verbraucht. Verwende die Option `--max-depth=1`, um nur die Verzeichnisse direkt in deinem Home anzuzeigen.

Auch wenn man mit der `--max-depth`-Option die Datenflut schon deutlich reduzieren kann, ist die Ausgabe gerade bei großen Verzeichnissen immer noch recht unübersichtlich. Um schnell große Dateien zu finden, wäre es also hilfreich, die vielen Informationen von `du` sortieren zu können. Filtere die Ausgabe von `du` durch `sort -n` (`-n` sortiert die Daten nicht einfach alphabetisch, sondern erkennt auch Zahlen)<sup>2</sup>. Wenn du willst, kannst du die sortierte Ausgabe noch durch ein weiteres Programm *pipen*, wie z. B. `tail` oder `less`.

Beachte: `sort` kann die „human-readable“-Ausgabe von `du` nicht korrekt sortieren!

Noch intuitiver ist das Tool `filelight`, welches die Größe von Verzeichnissen grafisch darstellt. Am besten ruft man `filelight` aus einer Shell auf und übergibt ihm gleich das Verzeichnis, das man untersuchen will, als Parameter (z. B. `~` für das Home-Verzeichnis).

Allerdings kann man sich nicht mehr an den PCs anmelden, wenn man sein Quota überschritten hat. In diesem Fall kann man mit `Strg+Alt+F1` auf ein anderes *virtuelles Terminal* wechseln und sich dort anmelden, nach erfolgreichem Löschen kommt man mit `Strg+Alt+F7` wieder zurück zur grafischen Oberfläche (Abmelden nicht vergessen!).

## Aufgabe 4 – *ssh* und *scp*

Oft genug kommt man in die Situation, dass man sich auf einem anderen Rechner einloggen muss, um dort zu arbeiten. Entweder weil dieser bestimmte Eigenschaften (*bessere/r Prozessor/installierte Programme/...*) erfüllt oder weil auf diesem Dateien oder Programme liegen, zu denen man sonst keinen Zugriff hat. Das Kommando, mit dem man sich auf einem anderen Rechner anmelden kann, kennst du bereits aus der Vorlesung.

---

<sup>2</sup>`sort` muss natürlich warten, bis alle Daten vorhanden sind, es kann also ein bisschen dauern.

- Melde dich mit Hilfe von `ssh` auf dem Rechner `fai05` an und schaue, was in `/var/tmp/vorkurs/` liegt.  
*Hinweis:* Du kommst wieder auf den lokalen Computer zurück, indem du `exit` eintippst.
- Lege auf **deinem** Rechner im Verzeichnis `/var/tmp/` ein neues Verzeichnis mit deinem Loginnamen an und kopiere (mittels `scp`) das, was du auf der `fai05` gefunden hast, in dieses Verzeichnis. (*Tipp: Die Rechnernamen stehen auf den Monitoren oder Rechnern*)
- Entpacke das Archiv, das du dir gerade kopiert hast, mit dem Befehl `unp`.

Einige der entpackten Dateien kannst du nicht betrachten, da du dazu nicht berechtigt bist. Lesbar machen kannst du sie mit `chmod` (siehe nächste Aufgabe).

## Aufgabe 5 – *chmod*

- Schau dich doch mal in den *Homes* der anderen Teilnehmer um.
- Wechsle mit dem Kommando `cd ..` in das Verzeichnis, in dem dein *Home* liegt und lass dir anzeigen, welche Unterverzeichnisse in diesem Verzeichnis liegen.
- Wechsle wieder mit `cd` in ein Verzeichnis deiner Wahl.
- Betrachte mit den dir mittlerweile bekannten Kommandos (`less`, `cat`) die Textdateien, die in diesem Verzeichnis liegen.
- Fällt dir etwas auf, wenn du die Datei `.bash_history` öffnen willst?  
(*Tipp: Schau dir die Rechte der Dateien mal etwas genauer an*)
- Versuche nun die Berechtigungen für die Dateien aus der vorherigen Aufgabe so zu setzen, dass du die Bilder anschauen und die Texte bearbeiten kannst.

## Aufgabe 6 – *Failed Backup*

Im Verzeichnis `/proj/ciptmp/vorkurs/uebung/aufgabe_failedbackup/archive/` findest du ein Archiv mit den Namen `failedbackup.tar.bz2`. In dieses Archiv wurden „versehentlich“ falsche Dateien eingepackt.

Kopiere das Archiv `failedbackup.tar.bz2` in ein temporäres Verzeichnis (z. B. `/var/tmp/<username>`) und entpacke es `dort`.

Wechsle nun in das entstandene Verzeichnis `failedbackup`.

- Wie viele Dateien sind im Verzeichnis `dir2`?<sup>3</sup> (Bitte nicht per Hand zählen!)<sup>4</sup>

Nun wollen wir das Backup mal bereinigen und ein paar Dateien löschen:

- Lösche alle Dateien, die mit `old_` beginnen, aus dem Verzeichnis `dir2`.

---

<sup>3</sup>Ich biete 1842

<sup>4</sup>Tipp: `ls` und `wc` geschickt „verbinden“

- Lösche alle Dateien, die mit `arts` beginnen, aus dem Verzeichnis `dir1`.<sup>5</sup>
- Lösche alle Dateien, die auf `.bak` enden, aus den *beiden* Verzeichnissen `dir1` und `dir2`.
- Wie viele Dateien sind nun noch in den beiden Verzeichnissen?<sup>6</sup>

Um herauszufinden, ob du die richtigen Dateien entfernt hast, kannst du dir eine Liste mit den Dateinamen erzeugen und eine Prüfsumme darüber berechnen lassen.

- Wechsle ins Verzeichnis `failedbackup`.
- Verwende `find`, um alle Dateien anzeigen zu lassen. Sortiere die Ausgabe mit `sort` (um gleiche Sortierung zu erreichen, verwende die Argumente `-fd`) und leite die Ausgabe an das Programm `md5sum`<sup>7</sup> weiter.

Die Aufgabe ist korrekt erledigt, wenn die Prüfsumme `a9fe1f26071319745682f0d717154db4` ist. Solltest du eine andere Prüfsumme haben, dann prüfe, ob noch zu löschende Dateien existieren. Oder hast du vielleicht eine neue Datei angelegt?

Erstelle nun aus dem Verzeichnis ein neues Archiv mit dem Namen `fixedbackup-ok.tar.bz2`.

## Aufgabe 7 – *Kill me!*

Führe zuerst das Programm `/proj/ciptmp/vorkurs/uebung/aufgabe_killme/killme_easymode` aus. Öffne danach eine neue Shell und versuche das Programm zu beenden.

War das nun zu leicht für dich, dann versuche dich am Programm `/proj/ciptmp/vorkurs/uebung/aufgabe_killme/killme`.

Und die Moral von der Geschichte<sup>7</sup>: Nicht einfach Programme von anderen Leuten ausführen, wenn man nicht weiß, was sie tun. ☺

## Aufgabe 8 – *Feedback*

Wir würden uns über Feedback zu unserem Einführungskurs freuen. Dazu gibt es unter <http://fsi.informatik.uni-erlangen.de/vorkurs/feedback> ein kleines Evaluationssystem, in dem jeder anonym seinen Kommentar zur Veranstaltung abgeben kann. Nimm dir bitte die Zeit und gib uns Feedback.

Anregungen für Feedback:

Wie hat dir die Vorlesung gefallen? Wie waren die Übungen? Welchen Eindruck haben die Übungsleiter auf dich gemacht? Könntest du den Ersties des nächsten Jahres den Kurs empfehlen?

## Zusatzaufgaben

Du hast noch Interesse an weiteren nützlichen Übungsaufgaben? Dann guck doch mal unter folgender Adresse vorbei: <https://fsi.informatik.uni-erlangen.de/dw/informationen/vorkurs/aufgabensammlung/>

---

<sup>5</sup>Keine Berechtigung? Da fehlen wohl irgendwelche Rechte. Nur welche?

<sup>6</sup>Ich biete 32 und 565

<sup>7</sup>`md5sum` liest von `stdin` Daten ein und bildet eine Prüfsumme über die Daten.