



Übungen zur Vorlesung
Werkzeuge der Informatik
– 2. Übung –

Datum: 18. 12. 2009

Aufgabe 1 (5 Punkte, Suchpfade)

1. Wechseln Sie in ihr Home-Verzeichnis. (d.h., dass danach ihr Home das *current working directory (CWD)* ist)
2. Starten Sie eine Bash (`bash`), und beenden Sie diese danach wieder mit dem Befehl `exit`.
3. Laden Sie die Datei `bash` von der Vorlesungs-Homepage herunter, und speichern Sie diese in ihrem Home-Verzeichnis.
4. Machen Sie diese Datei für sich ausführbar (executable). (Tip: `chmod`)
5. Geben Sie das Kommando an, mit dem Sie sicherstellen können, dass der Punkt (also das CWD) das erste Verzeichnis in Ihrer Umgebungsvariablen (environment-variable) `PATH` ist. (Sie können dabei davon ausgehen, dass `PATH` schon einen Wert hat.)
6. Geben Sie nun wieder das Kommando `bash` ein — was passiert, und wieso?
7. Setzen Sie nun die Umgebungsvariable `PATH` so, dass der Punkt als Pfad nicht mehr enthalten ist, alle anderen Pfade aber schon noch (hier hilft nur Copy-and-Paste).
8. Geben Sie nun wieder das Kommando `bash` ein — was passiert, und wieso?
9. Löschen Sie nun die Datei `bash` wieder aus Ihrem Home-Verzeichnis!

Punkte:

_____ von 29

Gruppe / Tutor:

Name(n) & Matr.-Nr.:

Aufgabe 2 (2 Punkte, Einfacher Chat)

Starten Sie in einem Terminal-Fenster den Befehl `cat -u >> ~/chat.txt`. Starten Sie in einem zweiten Fenster den Befehl `tail -f ~/chat.txt`.

Tippen Sie nun im ersten Fenster einige Zeilen beliebigen Text ein. Erklären Sie die Ausgabe im zweiten Fenster.

(Wer möchte, kann dies einmal mit einem Kommilitonen ausprobieren, indem sich einer der beiden auf dem Rechner des anderen einloggt per `ssh`; dann muss der "Chat"-Filename entsprechend modifiziert werden.)

Aufgabe 3 (2 Punkte, Prozesse)

Führen Sie in einer Shell `ps -aeH` aus.

- a) Interpretieren Sie die Ausgabe.
- b) Identifizieren Sie genau den Prozess, der die Shell darstellt, aus der heraus Sie das Kommando abgeschickt haben.

Aufgabe 4 (6 Punkte, Spaß mit Pipelines)

Die Internet-Movie-Database `imdb` enthält eine Liste der besten 250 Filme aller Zeiten (worüber man streiten kann). Wir haben diese Liste von dort abgespeichert und in eine ASCII-Datei mit folgendem Format konvertiert:

Abgabe:

6/7. 01. 2010
zu Beginn der Übung



Ranking:Rating:Movie-Title:Year:Votes

Laden Sie die Datei `top250` von der Vorlesungs-Homepage herunter und speichern sie diese in Ihrem Home-Verzeichnis.

- Geben Sie den 100-ten Film aus der Liste aus. (Tip: lesen Sie die Manpage von `head` und `tail`).
- Geben Sie die Titel der besten zehn Filme in alphabetischer Reihenfolge aus. (Tip: `cut` und `sort`)
- Geben Sie die Jahre aus, in denen es nur ein Film in die Top 250 geschafft hat. (Tip: `uniq`)
- Nennen Sie die Anzahl der Jahre, in denen es ein Film in die Top 250 geschafft hat. (Tip: `wc -l`)

Aufgabe 5 (12 Punkte, Noch mehr Spaß mit Pipelines)

- Wieviele Prozesse laufen gerade auf dem System? (Hinweis: `ps -e, wc`)
- Zeigen Sie alle User-Namen an, die gerade einen Prozess auf dieser Maschine laufen lassen. (Hinweis: `ps -ef, cut`).¹
- Wie kann man die erste Zeile ("PID ..." bzw. "UID ...") der Ausgabe der vorherigen Kommando-Pipeline verschwinden lassen? (Tip: `grep -v`)
- Entfernen Sie Duplikate aus der Ausgabe der vorherigen Kommando-Pipeline. (Hinweis: `sort, uniq`)
- Generieren Sie aus der vorherigen Ausgabe eine "Top-Ten-Liste" (d.h.: welche Benutzer lassen die meisten Prozesse laufen?) Hinweis: Sie benötigen eine Option bei `uniq` sowie zweimal das Kommando `sort`.
- Geben Sie mit dem Kommando `du` die sog. *disk-usage* (in kB) aller Files und Verzeichnisse in `/usr` aus. Dabei soll für ein Unterverzeichnis von `/usr` dessen gesamte Usage inklusive aller seiner Unterverzeichnisse aufsummiert werden. Sortieren Sie die Verzeichnisse nach Größe.

¹ Diese und die folgenden Teilaufgaben lassen sich wesentlich besser testen, wenn Sie sich per `ssh` auf `login.in.tu-clausthal.de` einloggen, weil dort oft viele User gleichzeitig eingeloggt sind.



1 Tips und Hinweise:

grep : Der Befehl **grep** dient dem Durchsuchen einer Datei nach einem Ausdruck. Die einfachste Form **grep** **STRING** **FILE** gibt alle Zeilen der Datei **FILE** aus, welche die Zeichenkette **STRING** enthalten. Wird keine Datei angegeben, liest **grep** von **STDIN**.

Lesen Sie sich die Manpage von **grep** durch, insbesondere die Optionen **-e**, **-i**, **-v** (Wenn dort von einem *pattern* die Rede ist, dann können Sie für unsere Zwecke einfach einen String annehmen.)

Falls Sie zu Hause kein Linux haben, können Sie die Manpage auch hier online lesen. (Wie im übrigen alle Manpages!)

cut : **cut --fields=N --delimiter=T** schneidet aus allen Zeilen einer übergebenen Datei bzw. **STDINs** bestimmte Felder aus. Als Trennzeichen zwischen den Feldern einer Zeile fungiert das Zeichen **T**. Die Option **--fields=N** gibt das/die gewünschte/n Feld/er an (hier Nummer **N**).

sort : Sortiert eine Datei (bzw. **STDIN**) zeilenweise alphabetisch. **sort** **FILE** gibt also die Datei **FILE** zeilenweise sortiert aus. Mit der Option **-n** sortiert **sort** numerisch (dabei nimmt **sort** an, dass die Zahl am Anfang der Zeile beginnt).

uniq : Zwei aufeinander folgende Zeilen werden miteinander verglichen; sind beide identisch, wird der Doppelgänger entfernt. (Tip: vorher sortieren)

du : Dient zur Ausgabe der sog. Disk-Usage einzelner Files und Verzeichnisse. Tip: starten Sie **du /usr/***, analysieren Sie die Ausgabe, und suchen Sie sich dann die sinnvolle Option aus der Manpage zum Lösen der Aufgabe.