

Einführung in die Informatik¹ (LV 3012)
WS 20/21

Übungsblatt 5 (2 Punkte)
Praktische Übungen

Diese Übung behandelt weitere Eigenschaften der Shell und geht auf zusätzliche UNIX-Kommandos ein. Benutzen Sie zur Bearbeitung die `man`-Pages, üben Sie ggf. auch außerhalb des Praktikums und nehmen Sie ruhig auch ein UNIX-Buch (z.B. das vom RRZN Hannover) zur Hilfe. Kommen Sie vorbereitet zur Übung (Grundlage: Die auf Übungsblatt 4 angegebenen Abschnitte von SelfLinux und vom Unix Manual)!

Aufgabe 5.1 (Jobs):

- (a) Starten Sie mit dem Kommando `xterm`² eine Shell in einem neuen Fenster. Was geschieht mit der "alten" Shell? Was geschieht, wenn Sie ein `&` an das Kommando anhängen (`xterm &`)?
- (b) Führen Sie ein Kommando aus, das die Eingabe der Shell bis zu seiner Beendigung blockiert (s. (a) oder z.B. `vi`, `cat` aus Standardeingabe). Mit `Strg-Z` (`Ctrl-Z` auf amerikanischen Tastaturen) können Sie den laufenden Prozess stoppen (Meldung [stopped]), mit `bg` dann in den "Hintergrund" schicken. Tun Sie dies mit mehreren Prozessen nacheinander und verwenden Sie die Befehle `jobs`, `Strg-Z`, `fg` und `bg`, um diese laufenden "Jobs" zu manipulieren. Achten Sie auf die Job-Nummern, die Sie beim Starten eines Hintergrund-Jobs erhalten und verwenden Sie diese als Referenzen.
- (c) Beenden Sie einige der in (b) gestarteten Prozesse, indem Sie sie in den Vordergrund holen und regulär beenden (z.B. `exit` in einer Shell).

Aufgabe 5.2 (History):

Die zuletzt eingegebenen Kommandozeilen werden von der Shell in einer sog. *history* gespeichert. Sie können wiederholt werden, um sich das erneute Eingeben längerer Kommandos zu sparen. Für weitergehende Editiermöglichkeiten der aktuellen Eingabezeile sei auf die `man`-Pages oder die Literatur verwiesen, z.B. hier: c't 20/20, S.170ff.

- (a) Wie zeigt man die gespeicherten Kommandos an?
- (b) Wie wählt man ein bestimmtes Kommando zur Wiederholung aus?

¹ basierend auf der Veranstaltung von Prof. Dr. Reinhold Kröger & Ergänzungen von Prof. Dr. Martin Gergeleit

² Falls `xterm` nicht verfügbar, verwenden Sie eine Alternative. Unter Ubuntu bietet sich `gnome-terminal` an. Falls Sie keinen X-Server betreiben können, ist dieser Aufgabenteil nicht zu bearbeiten. Bitte vermerken Sie dies im Protokoll. Analoges gilt für weitere Aufgaben, die `xterm` verwenden.

(c) Wie kann am einfachsten das letzte Kommando wiederholt werden?

Aufgabe 5.3 (Shell-, Umgebungsvariablen):

Das Verhalten der Shell und die Konfiguration vieler Kommandos und Anwendungen werden durch Shell-Variablen bzw. sogenannte Umgebungsvariablen ("Environment Variables") beeinflusst, die Zeichenketten der Form `<Name>=<Wert>` sind und definiert, gesetzt und gelesen werden können. Hierbei bestehen Unterschiede zwischen `bash` und `(t)csch`. Im Folgenden wird von der Nutzung der `bash` ausgegangen.

(a) Lassen Sie sich die momentan gesetzten Shell-Variablen anzeigen.

(b) Finden Sie aus der Ausgabe zu (a) heraus, welche Shell Sie benutzen.

(c) Der Wert der Variablen `HOME` sollte Ihnen bekannt vorkommen. Was bedeutet er?

(d) Was besagt die Variable `PATH`? Hinweis: Listen Sie den Inhalt der in `PATH` angegebenen Verzeichnisse auf. Was finden Sie dort?

(e) Definieren Sie eine Shell-Variable `MYVAR` und weisen Sie ihr den Wert `"date"` zu.

(f) Lassen Sie sich mittels `echo` den Wert der Variablen `MYVAR` anzeigen.

(g) Wie kann man die Variable `PATH` um ein weiteres Suchverzeichnis erweitern?

(h) Legen Sie (wenn nicht bereits geschehen) ein Verzeichnis `bin` direkt unterhalb Ihres Homeverzeichnisses an und erweitern Sie `PATH` so, dass auch dieses Verzeichnis durchsucht wird. Wofür ist dieses Verzeichnis nützlich?

(i) Was bedeutet `eval $MYVAR`?

(j) Wie kann man die Definition der Variablen `MYVAR` zurücknehmen (löschen)?

(k) Wie zeigt man die definierten Umgebungsvariablen mit ihren Werten an?

(l) Definieren Sie eine Umgebungsvariable `MYENVVAR` mit dem Wert `"/usr/local/bin"`. Starten Sie `xterm &` und sehen Sie sich im neuen Fenster die hier definierten Umgebungs- und Shell-Variablen an. Was stellen Sie fest?

Aufgabe 5.4 (Quoting von Metazeichen, Kommandoersetzung):

Vor der Ausführung eines Kommandos wird die Eingabezeile von der *Shell* vorverarbeitet. Bestimmte Zeichen haben dabei eine besondere Bedeutung, wie &, |, ;, *, ?, usw.. Man nennt diese Zeichen *Metazeichen*. Um solche Zeichen oder ein Leerzeichen in Zeichenketten aufzunehmen, wird das sogenannte *Quoting* erforderlich.

- (a) Wie weist man der Shell-Variablen MYVAR den Wert "ls -l" zu?
- (b) Worin unterscheidet sich die Verwendung von " " und ' '? Was ist die Ausgabe von folgenden Zeilen und warum?
- ```
MY1=ist
MY2="heute $MY1 \"\$\"-Tag"
MY3='heute $MY1 \"\$\"-Tag'
echo $MY2
echo $MY3
```
- (c) ★ Ein besonders mächtiger Mechanismus der Shells ist die *command substitution*. Machen Sie einen ersten Versuch, indem Sie den Ausdruck `date` (in sog. *backquotes*, frz. *accent grave*) in eine `echo`-Zeile einbetten. Experimentieren Sie mit Kommandos, die ihre Parameter aus der Auswertung eines anderen Kommandos erhalten. Was kann in diesem Zusammenhang `cat `ls` > datei` bedeuten, wenn man annimmt, dass im aktuellen Verzeichnis nur Dateien und keine Unterverzeichnisse existieren?

#### Aufgabe 5.5: (Shell-Prozeduren)

Eine Folge von Kommandos kann in einer Datei abgespeichert und als Ganzes ausgeführt werden (Shell-Skript).

- (a) Legen Sie eine Datei `scr` an, die in aufeinanderfolgenden Zeilen die Kommandos `date`, `pwd` und `ls` enthält. Übergeben Sie die Datei an `sh` zur Ausführung. Können Sie sich die Ausgabe erklären?
- (b) ★ Ein Shell-Skript erlaubt ähnlich wie ein C-Programm auch Kontrollstrukturen zur bedingten Ausführung von Anweisungen, für Schleifen usw. Ermitteln Sie, welche Kontrollstrukturen es insgesamt gibt. Experimentieren Sie damit.

#### Aufgabe 5.6: (chmod, chown, chgrp):

- (a) Welche Rechte unterscheidet UNIX an Dateien und Verzeichnissen?
- (b) Schauen Sie sich Homeverzeichnis (ersatzweise: `~werntges`) an und erklären Sie die verschiedenen Zugriffsrechte bzgl. der einzelnen Dateien und Verzeichnisse.
- (c) Was bedeutet das x-Recht bei Verzeichnissen?
- (d) Ändern Sie die Zugriffsrechte der Datei `scr` aus Aufgabe 5.5 (a) so ab, dass Sie die Datei durch Aufruf von `scr` ausführen können.
- (e) Finden Sie mittels `whoami` bzw. `id` heraus, mit welchem Benutzernamen Sie arbeiten und welchen Gruppen Sie angehören.

- (f) Erzeugen Sie eigene Dateien bzw. Verzeichnisse mit unterschiedlichen Zugriffsrechten. Diese Aufgabe lässt sich besonders gut zu zweit erarbeiten, da so die Zugriffsrechte gegenseitig überprüft werden können.

#### Aufgabe 5.7: (ps, kill):

Die Ausführung jedes Programms und damit auch jedes Kommandos geschieht als sogenannter Prozess.

- (a) Zeigen Sie alle Ihre laufenden Prozesse an. Probieren Sie einige Anzeige-Optionen aus.
- (b) Starten Sie in einem neuen Fenster den Editor vi für eine neue Datei. Finden Sie heraus, welcher Prozess das gestartete Programm vi ausführt.
- (c) Mittels `kill <pid>` wird dem Prozess mit dem process identifier *pid* ein Signal geschickt, auf das hin er vom Betriebssystem beendet wird. (Diese Vorgehensweise wendet man gelegentlich an, wenn ein Prozess "hängt" und sich anderweitig nicht mehr ansprechen lässt. Testen Sie, indem Sie einen Prozess für den PDF-Reader (z.B. *evince*) starten und diesem das Terminierungssignal SIGTERM senden.). Was beobachten Sie? Listen Sie erneut alle Prozesse.

#### Aufgabe 5.8: (find):

`find` ist ein sehr mächtiger Befehl, mit dem im Dateisystem bestimmte Dateien abhängig von Suchkriterien gefunden werden können.

- (a) Versuchen Sie, folgenden Befehl zu verstehen: `find /usr -name "xterm"`
- (b) ★ Listen Sie mit einem Befehl alle Dateien unter Ihrem Home-Verzeichnis, die seit 2 Wochen nicht mehr verändert wurden. Wie lautet dieser?

#### Aufgabe 5.9:

★ (optional) Wechseldatenträger wie CD-ROMs und USB-Sticks dürfen im Gegensatz zu normalen Laufwerken meist auch von Anwendern „gemountet“ werden. Auf den Praktikumsrechnern der Hochschule erkennen Sie diese am Schlüsselwort `user` in der vierten Spalte der Datei `/etc/fstab`. Finden Sie dort mit `grep` heraus, welche dies auf Ihrem Rechner sind, und über welche „mountpoints“ Sie auf diese Datenträger ggf. zugreifen können. – Nur auf dem eigenen Rechner: Verfolgen Sie die neuen Meldungen in `/var/log/syslog` (vormals: `/var/log/messages`) bei dem Einstecken eines USB-Sticks. Als welches „Gerät“ erscheint der Stick?

#### **Vorbereitungen für Übungsblatt 06:**

- Wiederholen Sie bzw. erarbeiten Sie sich HTML5-Grundlagen. Bearbeiten Sie dazu das folgende Tutorial: <https://www.w3schools.com/html/default.asp> (HTML Tutorial) bis einschließlich Abschnitt „HTML Block & Inline“.
- Kommen Sie vorbereitet in die Übung – für Übung 6 ist dies besonders wichtig!