

Anmerkungen zu UNIX

Auf den Rechnern, die für das FORTRAN-Praktikum benutzt werden, ist das Betriebssystem UNIX (genauer: die Variante LINUX) installiert. Darüber hinaus kann jeder Benutzer unter einer Anzahl von UNIX-Erweiterungen wählen, den sog. *shells*, die mehr oder weniger weit verbreite Standards in der UNIX-Welt sind. Überall vorhanden ist die Bourne-shell und meist stehen auch die C- und K-shell zur Verfügung (LINUX-Standardshell: `bash` – Bourne-again-shell). Die *shells* unterscheiden sich vor allem in der Benutzerfreundlichkeit, (leider) der Variablensyntax sowie den zur Verfügung stehenden UNIX-Kommandos. Um auch die Möglichkeiten moderner X-Terminals zu nutzen, wird der Aufbau einer Bildschirmseite vom sog. *window manager* gesteuert (CIP-Pool: FVWM oder KDE). Dieser erlaubt beispielsweise das Öffnen und Schließen oder das Verschieben eines Fensters. Zu Beginn einer Sitzung am Terminal existieren normalerweise entsprechend der benutzten *shell* und des *window managers* Voreinstellungen, die einen Standard-Bildschirmaufbau vollziehen und später vom Benutzer bei guter Kenntnis des Systems selbst verändert werden können.

Im folgenden soll auf einige Anwendungen eingegangen werden. Weitere Details können beispielsweise im UNIX-*manual* des Regionalen Rechenzentrums für Niedersachsen (RRZN) nachgelesen werden, das für 7.- DM im Rechenzentrum (Zi. 1.023, Beratung) bezogen werden kann (Stand: 1999).

Wie loggt man sich ein?

1. Evt. booten und Linux auswählen
2. Im Login-Fenster **USER-ID** (z.B. `snhafran`) + **Return-Taste**
3. **Passwort** + **Return-Taste** Alle Kommandos oder Eingaben werden mit der Return-Taste abgeschlossen (im folgenden nicht mehr ausdrücklich erwähnt).

Wie bekomme ich Information über UNIX-Kommandos? UNIX-Befehle bestehen aus dem eigentlichen Befehl (z.B.: `ls`), den Optionen (z.B.: `-lai`) und einem Objekt (zumeist eine Datei), auf das der Befehl angewendet wird (z.B.: `l*`). Um mehr Information über Syntax und Optionen sowie weitere im Zusammenhang stehende Befehle zu bekommen, informiere man sich über die **man-pages** (auf LINUX-Rechnern alternativ durch **info**) beispielsweise mit **man** (Kommando) (z.B.: `man ls` bzw. `info ls`), oder man benutze die (gedruckten) *manuals*.

Wie ändert man das Passwort? `passwd`

Wie erstelle ich eine Datei? Textdateien (im folgenden werden Dateien wie oft üblich auch als *files* bezeichnet) wie z.B. FORTRAN-Quelltexte werden meistens mittels eines Editor-Programms erzeugt. Auf den allermeisten UNIX-Systemen steht hierfür der **emacs**-Editor oder Varianten (**xemacs**, **jed**) zur Verfügung (Aufruf z.B.: `emacs test.txt`). `emacs` ist in vielen UNIX-Büchern (z.B. "Linux in a nutshell") beschrieben (wichtigste Tastenkombination: (`Ctrl-x Ctrl-c`) um den Editor wieder zu verlassen).

Aufgrund seiner generellen Verfügbarkeit (quasi Teil des Betriebssystems) sei auch alternativ der **vi**-Editor erwähnt, der allerdings etwas gewöhnungsbedürftig ist (Aufruf z.B.: `vi test.txt`). Der **vi**-Editor unterscheidet zwei Modi: a) den Kommandomodus, in dem man beispielsweise Zeichen löschen kann (`x`) oder den Editor mit (`:wq!`) oder ohne (`:q!`) Abspeichern verlassen kann. Zeilen können mit `dd` gelöscht werden. Und b) den Textmodus, in dem man Texte erstellen kann. Den Textmodus kann man mit der *escape*-Taste (`Esc`) verlassen, während man ihn mittels *insert* (`i`) oder *append* (`a`) aus dem Kommandomodus erreicht. Weitere Details sind im Manual bzw. den *man-pages* zu finden.

Wie kann man *files* bearbeiten?

- `lp -dlj4 test.txt` Ausdrucken eines *files* auf Drucker `lj4`
- `cp test.txt test1.txt` Kopieren eines *files*

- `mv test.txt test1.txt` Umbenennen eines *files*
- `rm test.txt` Löschen eines *files*
- `cat test.txt` Auslisten eines *files* am Bildschirm
- `chmod go+rw test.txt` Setzen von Zugriffsrechten (Den Benutzern, die zur Gruppe *group* (g) und der restlichen Welt (o) gehören, werden die Rechte für Lesen (r), Schreiben (w) und Ausführen (x) der Datei test.txt eingeräumt; s.a. `ls`-Kommando).

Wie arbeitet man mit Dateienverzeichnissen (im folgenden *directory*)? *Directories* sind in einer Baumstruktur miteinander verkettet. Alle Kommandos wirken nur auf das aktuelle Verzeichnis, wenn nicht explizit ein anderes Verzeichnis angegeben wird.

- `ls` Inhalt der aktuellen *directory* (mit `ls -lai` erhält man ein ausführliches *listing*, das Zugriffsrechte, Besitzer, Länge und Erstellungsdatum der *files* beinhaltet.)
- `pwd` Pfadname des aktuellen *directory*
- `mkdir testdir` Erzeugen einer *directory*
- `rmdir testdir` Löschen einer *directory*
- `cd testdir` Wechseln der aktuellen *directory*
(Auf die darüberliegende Ebene kommt man z.B. mit `cd ..`, auf das Ausgangsverzeichnis mit `cd` oder mit dem absoluten Pfadnamen, z.B. `cd /home/snhafran/aufgabe1`. Komplette file-Namen beinhalten übrigens auch den Pfad!)

Wie kann ich ein lauffähiges FORTRAN-Programm erzeugen?

1. `f77 test.f` erzeugt ein lauffähiges Programm *a.out*.
2. *a.out* Ausführen des Programms *a.out*
3. `f77 -c test.f` Nur Compilieren des Programms
4. `f77 -o hallo hallo.f` erzeugt das lauffähige Programm *hallo*

Sonstiges

- In jedem Fall ist Groß- und Kleinschreibung signifikant.
- Ausgaben auf den Bildschirm können in *files* umgelenkt werden: `ls > list.txt` schreibt das Inhaltsverzeichnis in das file *list.txt*. Ähnlich können Bildschirmausgaben aus FORTRAN-Programmen umgeleitet werden (`a.out > out.dat`).
- Joker können verwendet werden. Z.B. listet `ls *.f` alle files im aktuellen *directory* mit der Endung *.f* auf.
- Die Ausgabe eines Befehls kann als Eingabe eines nächsten Befehls dienen – sog. *pipelining* –; dies erfolgt durch das Zeichen `|`. Ein Anwendung hierfür ist das seitenweise Darstellen eines files `cat test.txt | more`. Der `cat`-Befehl gibt den Inhalt des *files* aus und *pipet* ihn weiter an das Kommando `more`, das die seitenweise Ausgabe am Bildschirm durchführt.
- Gewisse Dateinamen sollten entsprechend der Namenskonventionen benannt werden; dies ist nicht vorgeschrieben, erleichtert aber manche Arbeitsgänge. So sollten FORTRAN-Quelltexte auf *.f* enden, Textdateien auf *.txt*. Der FORTRAN-Compiler erzeugt Dateien mit der Endung *.o*. *Files*, die mit einem Punkt beginnen, sind meist Parameter- oder Systemdateien. Solche Dateinamen sollte man daher ebenso meiden wie die Benutzung von Sonderzeichen im Dateinamen (z.B. `/,*,?...).`
- Es besteht die Möglichkeit, anderen Benutzern Nachrichten über *electronic mail* (kurz *email*) zukommen zu lassen, z.B.: wird mit
`mail napiwotzki@sternwarte.uni-erlangen.de < message.txt` das *file* *message.txt* an den Benutzer `napiwotzki@sternwarte.uni-erlangen.de` geschickt. Durch den Aufruf `mail` läßt sich angekommene *mail* lesen.
- **exit** Schließen eines Fensters bzw. Beenden der Sitzung (**solange, bis die Meldung fürs Einloggen erscheint**, gegebenenfalls über den *window-manager*).