Einige wichtige UNIX-Kommandos *:

Alle Befehle unter Unix sind grundsätzlich durch Betätigen der Return-Taste nach Eingabe des Befehls "abzuschicken". (Diese Taste wird oft durch `<cr>` für "carriage return" bezeichnet.)

`passwd` oder `yppasswd`	–	zum Setzen oder Ändern des benutzereigenen Paßwortes.
`logout` oder `exit`	–	zum Verlassen des Systems.
`C-d`	–	äquivalent zu `logout` (gleichzeitiges Betätigen der Ctrl-Taste und der Taste mit dem Buchstaben d).
`ls`	–	zeigt dem Benutzer seine schon vorhandenen abgespeicherten Dateien.
`ls -l`	–	wie `ls`. Gibt aber zusätzliche Informationen ueber die Dateien wie z. B. die Größe in Bytes und das Datum der letzten Veränderung.
`more Dateiname`	–	druckt den Inhalt der Datei mit dem Namen `Dateiname` auf dem Bildschirm aus.
`cp Dateiname1 Dateiname2`	–	kopiert die Datei mit dem Namen `Dateiname1` in die Datei mit dem Namen `Dateiname2`. Vorsicht! Wenn schon eine Datei mit Namen `Dateiname2` vorhanden ist, so wird sie durch dieses Kommando überschrieben.
`mv Dateiname1 Dateiname2`	–	benennt die Datei mit dem Namen `Dateiname1` um in `Dateiname2`. Vorsicht! Wenn schon eine Datei mit Namen `Dateiname2` vorhanden ist, so wird sie durch dieses Kommando überschrieben.
`rm Dateiname`	–	löscht die Datei mit dem Namen `Dateiname`.
`emacs Dateiname`	–	Aufruf des Texteditors EMACS, um den Text in der Datei mit Namen `Dateiname` zu bearbeiten. Zum Beispiel können Sie einen neuen Programmtext schreiben oder einen schon vorhandenen abändern.
`gcc -o Name1 Name2.c`	–	compiliert (übersetzt) die Datei mit Namen `Name2.c`, in der als Text ein C-Programm steht, und legt das ausführbare Resultat in der Datei `Name1` ab. Beachten Sie, daß der Name der Datei, in der das C-Programm steht, mit `.c` enden muß. Das übersetzte Programm können Sie nun ausführen, indem Sie `./Name1` eingeben.

* Material zu diesem Kurs im World Wide Web unter der Adresse
`http://math36.mathematik.uni-bielefeld.de/Ckurs-a/`

`gcc Name.c`	–	compiliert die Datei mit Namen `Name.c`, falls sie ein C-Programm enthält, und legt das ausführbare Resultat in der Datei mit dem Standard-Namen `a.out` ab. Durch anschließende Eingabe von `./a.out` kann man das übersetzte Programm ausführen.
`man Kommando`	–	liefert den Manual-Eintrag für den Unix-Befehl namens "`Kommando`"
`man -k Keyword`	–	liefert Kurzbeschreibungen von Kommandos, die im Zusammenhang mit dem Stichwort "`Keyword`" stehen; z. B. liefert `man -k copy` eine Übersicht über Unix-Kommandos, die mit "`copy''` zu tun haben.
`apropos Keyword`		ist ein zu `man -k Keyword` gleichwertiger Befehl.
`mkdir C-progs`	–	erstellt ein Verzeichnis oder ein Directory mit Namen `C-progs`, der Name kann frei gewählt werden.
`rmdir C-progs`	–	löscht ein Verzeichnis oder ein Directory mit Namen `C-progs`.
`cd C-progs`	–	macht das Verzeichnis `C-progs` das aktuelle Verzeichnis
`pwd`	–	gibt den Namen des "aktuellen" Verzeichnisses zurück.

Die Verzeichnisse in Unix sind sind Datenbereiche, in denen mehrere Dateien oder auch Unterverzeichnisse angelegt und verwaltet werden können. Sie sind "baumartig angeordnet", d. h. jedes Verzeichnis kann Unterverzeichnisse enthalten und ist in genau einem übergeordneten Verzeichnis enthalten, abgesehen von dem obersten Verzeichnis des Systems, dass den Namen `/` hat und Root-Verzeichnis genannt wird. Eine Abkürzung für den Namen des jeweils aktuellen Verzeichnisses ist "`.` ", eine Abkürzung für das jeweils übergeordnete Verzeichnis ist "`..`". Der Befehl `cd ..` macht also das übergeordnete Verzeichnis "aktuell". Jeder Benutzer hat ein "Home"-Verzeichnis, in dem er seine Daten erstellt. Für den Benutzer `cr00` dieses Kurses z. B. ist das

`/amnt/mcl04/users/sdb1/cr00.`

Eine Abkürzung für das eigene Home-Verzeichnis ist `~` .

Übungsaufgabe 1: Welche Resultate haben die folgenden Unix-Befehle:

`cd .`   `cd ..`   `cd /`   `cd ~`

Überprüfen Sie jeden Befehl, indem Sie anschließend die Befehle `pwd` und `ls -l` geben.

Übungsaufgabe 2: Studieren Sie die Manual-Einträge aller bisher besprochenen Befehle, indem Sie `man ls, man cp, man passwd` usw. eingeben und die Ausgabe lesen. Was tut z. B. der Befehl "`cp -r dir1 dir2`"?

Wichtige Editor-Kommandos von EMACS:

Der Editor EMACS wird gestartet, indem man

```
emacs Dateiname &
```

eingibt, wenn das UNIX-Betriebssystem Kommandos erwartet. `Dateiname` ist der Name der Datei, welche bearbeitet werden soll. EMACS startet, indem ein eigenes Fenster geöffnet und der Cursor (die Schreibmarke) in die linke obere Ecke positioniert wird. Nun können Sie Text eintippen. Das Zeichen & am Ende des Befehls bedeutet, daß das Unix-Fenster sofort wieder für neue Kommandos bereit ist.
Die folgenden EMACS-Kommandos können mit den angegebenen Tastenbefehlen oder aber auch durch "Mausklick" über die Kopfzeile im EMACS-Fenster (zu Beginn kann gewählt werden: `Buffers Files Tools Edit ...`) ausgeführt werden.
EMACS wird verlassen, indem Sie

`C-x` – (gleichzeitig **Ctrl**- und x-Taste drücken)

und danach

`C-c` – (gleichzeitig **Ctrl**- und c-Taste drücken)

eintippen. Dies bewirkt, daß der Text nach Rückfrage durch EMACS (in der letzten Zeile) und Bestätigung durch Eingabe von `y` in die Datei `Dateiname` abgespeichert und EMACS beendet wird. Sie sichern (speichern) ihre Datei, indem Sie

`C-x C-s`

eingeben. EMACS wird dabei nicht verlassen, so daß Sie sofort weiterarbeiten können. Sie können mit den Pfeiltasten der Tastatur im schon geschriebenen Text herumwandern und Zeichen einfügen oder löschen.

`C-d` – ("**d**elete") löscht das Zeichen unter dem Cursor.
`C-k` – ("**k**ill") löscht die aktuelle Cursorzeile von der Position des Cursors bis zum Ende.
`C-y` – ("**y**ank") macht die letzte Löschung wieder rückgängig.

Sie können in eine bestimmte Zeile des Textes springen, wenn Sie die Zeilennummer dieser Zeile wissen. Dies geschieht mit folgendem Verfahren: Geben Sie die Tastenfolge

`C-[ x` oder `ESC x`

ein. Der Cursor springt nun in die letzte Zeile des Bildschirms. Dort steht dann die Zeichenfolge `M-x`. Geben Sie jetzt

```
goto-line <cr>
```

ein. In der letzten Zeile erscheint nun `Goto line:`, und Sie können die Zeilennummer der gewünschten Zeile, gefolgt von <cr>, eingeben.
Eine Automatisierung dieser Prozedur finden Sie weiter unten beschrieben.
Die "ESC"-Taste (ESC steht für "Escape") ist (oft) die Taste F11 in der obersten Reihe der Tastatur. Äquivalent zu ihr ist `C-[`. Sie wird manchmal als Metataste bezeichnet.
Bewegungen des Cursors (= Schreibmarke), im Text:

`C-a` – zum **A**nfang der Cursorzeile
`C-e` – zum **E**nde der Cursorzeile
`ESC f` – ("**F**orward)" zum nächsten Wort
`ESC b` – ("**B**ackward)" ein Wort zurück
`C-v` – ("**V**ertical)" zum nächsten Bildschirmausschnitt im Text
`ESC v` – ein Bildschirmausschnitt zurück

Suchen im Text:

C-s	–	suche nach einer bestimmten Zeichenfolge von der Cursorposition bis zum Textende. Die Zeichenfolge wird auf der letzten Zeile des Bildschirms, der Kommandozeile von EMACS, vom Benutzer eingegeben. Beendet wird die Suche mit ESC oder C-g. Mit diesem Tastendruck (C-g) kann jeder EMACS-Befehl abgebrochen werden.
C-r	–	suche rückwärts von der Cursorposition bis zum Textbeginn. Sonst wie C-s.

Bei Fehlern, die Sie gemacht haben, helfen eventuell folgende Kommandos:

C-g	–	unterbricht ein an EMACS gegebenes Kommando
C-x u	–	macht das letzte (Lösch-)Kommando rückgängig, dies kann für das vorletzte Kommando wiederholt werden, usw.

Sie können eine andere Datei in den Editor laden, indem Sie

```
C-x C-f
```

eingeben. In der letzten Zeile des Bildschirms werden Sie dann nach dem Namen der neuen Datei gefragt. Tippen Sie ihn gefolgt von <cr> ein. EMACS eröffnet nun ein neues Fenster bzw. einen neuen “buffer” für diese Datei, und Sie können mit diesem weiterarbeiten.
Wenn Sie kein neues Fenster bzw. neuen “buffer” bei dieser Operation haben wollen, müssen Sie die neue Datei mit

```
C-x C-v
```

laden. Wieder fragt Sie EMACS nach dem Namen der neuen Datei. Er lädt sie nach Angabe des Namens ein, vergißt aber alles, was in Ihrem jetzigen “buffer” steht.

Das .emacs-File und weitere Informationen über EMACS.

Die Arbeit mit EMACS kann mittels einer Datei namens .emacs im Hauptverzeichnis vereinfacht werden. Diese Datei wird von EMACS beim Start gelesen und EMACS wird entsprechend konfiguriert.
Einige Beispiele: Wenn Sie die Zeile

```
(global-set-key "\C-cg " 'goto-line)
```

in diese Datei einfügen, wird damit die Funktion “goto-line” auf die Tastenfolge “C-c g” gelegt. Nach Eingabe dieses Befehls fragt EMACS nach einer Zeilennummer, nach deren Eingabe er in diese Zeile springt.
Das Einfügen der Zeilen

```
(setq text-mode-hook
  '(lambda () (auto-fill-mode 1)))
```

bewirkt, das EMACS grundsätzlich in den “Auto-fill-mode” geht, d. h. daß Zeilen beim Edieren automatisch umgebrochen werden.
Durch Eingabe der Tastenfolge C-h kann man “On line-Hilfe” bekommen. Es gibt sogar ein Tutorial. Die weitere Eingabe eines Fragezeichens “?” liefert genauere Hinweise.