

Einführung in Emacs

Computerlinguistik der Universität Erlangen
zusammengestellt von Martin Lorenz
überarbeitet von Matthias Bethke und Jörg Kapfer

1. Februar 2007

Inhaltsverzeichnis

1 Was ist Emacs?

Emacs ist ein *Editor*, also ein Programm zum Eingeben und Verändern von Texten. Im Gegensatz zu *Textverarbeitungsprogrammen* sind die Texte, die mit Editoren eingegeben werden, nicht in erster Linie zum Ausdrucken gedacht. Verschiedene Schriftarten und -größen, Proportionschrift, Fettdruck, automatische Paginierung (Aufteilen des Textes in Seiten) etc. sind nicht Sache eines Editors. Dafür gibt es Programme wie L^AT_EX oder OpenOffice.org. Die Texte, die mit einem Editor eingetippt werden, sind vor allem Programme (in den unterschiedlichsten Programmiersprachen) und ähnliche Texte, die für die Steuerung des Computers benötigt werden.

Der Editor Emacs unterstützt viele verschiedene Programmiersprachen mit Spezialbefehlen, die zum Beispiel ganze Anweisungen auf einen Tastendruck in das Programm einsetzen oder Programmzeilen automatisch hübsch einrücken.

Um den unterschiedlichen Wünschen von vielen Benutzern gerecht zu werden, wurde in Emacs die Programmiersprache ELISP eingebaut¹, die an LISP angelehnt ist. Damit ist es möglich, seine eigene Unterstützung für neue Programmiersprachen zu schreiben (in der Erlanger Computerlinguistik existiert etwa ein eigener Modus für MALAGA), oder sogar Spiele u.ä. zu schreiben.

Der Name des Editors setzt sich aus „**E**ditor with **M**acros“ zusammen. Makros sind Folgen von Tastendrücken – z. B. unterschiedlichen Edierbefehlen – die zusammengefasst werden und mit einem einzigen Tastendruck wieder abgspult werden können. Emacs verfügt selbstverständlich auch über diese Möglichkeit.

2 Wie startet man Emacs?

Emacs kann von einer Shell aus mit dem Befehl „emacs“ gestartet werden. Wenn man schon weiß, welche Datei man edieren möchte, kann man den Namen dieser Datei als Argument angeben. Wer etwa die Datei „text“ eingeben oder verändern möchte, gibt „emacs text“ ein. Dann erscheint der Text sofort auf dem Bildschirm, sobald Emacs gestartet wurde. Es ist auch möglich, mehrere Textdateien beim Starten anzugeben; dann werden die Dateien in getrennten Emacs-Fenstern übereinander gezeigt.

Wie (fast) alle anderen Programme kann auch Emacs mit dem nachgestellten Kaufmannsund als Hintergrundbefehl gestartet werden, wenn die Shell in einem X-Terminal läuft. Auf der Textkonsole funktioniert Emacs genauso, nur wird hier natürlich kein eigenes Fenster geöffnet, sondern der Editor muss als Vordergrundprozess laufen.

¹Tatsächlich ist der größte Teil von Emacs selbst in dieser Programmiersprache geschrieben.

3 Tastatur-Befehle

Viele häufig verwendete Emacs-Befehle können über eine Kombination von Befehlstasten eingegeben werden. In dieser Übersicht werden die Befehlstasten durch eine Zeichenkombination wie C-x oder M-s dargestellt. Eine Kombination, die mit C beginnt, wird eingegeben, indem die **[Strg]**-Taste gehalten wird und das Zeichen, das hinter dem Bindestrich steht, eingetippt wird. Bei einer Kombination, die mit M beginnt, muss die Taste **[Alt]** gehalten und die entsprechende Zeichentaste gedrückt werden².


Wenn man einen Tastatur-Befehl noch nicht vollständig eingegeben hat, oder wenn ein Befehl abgebrochen werden soll, dann kann man C-g eintippen. Manchmal muss man das zweimal tun, aber spätestens dann erscheint in der untersten Zeile des Emacs-Fensters ein „Quit“ als Bestätigung, dass der Befehl abgebrochen wurde.

4 Text eingeben und ändern

Wenn Emacs mit einem Dateinamen gestartet wurde, wird der Text in dieser Datei automatisch angezeigt, sobald das Emacs-Fenster geöffnet wurde.

Wenn man keinen Dateinamen angegeben hat, stehen nach dem Start ein paar Hinweise zur neuesten Version des Emacs auf dem Bildschirm. Sie verschwinden, sobald man eine Taste drückt, und es erscheint ein leeres Fenster, in das man Text eingeben kann.

In Emacs ist normalerweise der *Einfügemodus* eingestellt. Das heißt, man kann wie auf einer Schreibmaschine Text eintippen. Die eingegebenen Zeichen erscheinen dort, wo das kleine Rechteck (der *Cursor*) im Emacs-Fenster zu sehen ist. Ist kein Cursor zu sehen, dann liegt das meistens daran, dass das Emacs-Fenster nicht aktiviert ist.

Der Cursor springt an den Anfang der nächsten Zeile, wenn man die Return-Taste () drückt. Sie tut also das gleiche wie der Wagenrücklauf auf der Schreibmaschine. Hat man sich einmal verschrieben, so kann man das letzte eingegebene Zeichen mit der Backspace-Taste wieder löschen. Auch die Wirkung der Return-Taste wird mit der Backspace-Taste wieder rückgängig gemacht.

Häufig möchte man in bereits eingetipptem Text wieder etwas löschen, verändern oder wieder einfügen. Dazu muss man natürlich erst einmal an die richtige Stelle kommen. Das geschieht mit den *Cursortasten*, das sind die Tasten mit den Pfeilen, die auf der Tastatur rechts unterhalb der Buchstabentasten liegen. Mit ihnen kann man den Cursor zeichenweise in „alle vier Himmelsrichtungen“ bewegen.

Wenn man dann an der richtigen Stelle ist, wo man z. B. einen Teil des Textes löschen möchte, kann man entweder die Backspace-Taste benutzen (sie löscht alles zeichenweise, was *vor* dem Cursor steht), oder die Entf-Taste (sie löscht Zeichen für Zeichen *hinter* dem Cursor). Tippt man Zeichen ein, während der Cursor mitten in einem Text steht, dann werden diese Zeichen in den Text eingefügt: der Textteil, der hinter dem Cursor steht, wird nach hinten verschoben.

5 Laden und Speichern

Ein Text, den man in Emacs eingibt, wird im Hauptspeicher des Rechners abgelegt, d.h. wenn der Editor beendet wird (sei es, weil man sich ausloggt, der Strom ausfällt oder der Rechner abstürzt) gehen diese Daten verloren. Deshalb sollte man den Text ab und zu abspeichern. Das macht man mit dem Befehl C-x C-s (für *save*). Wenn der Text das erste Mal gespeichert wird, fragt Emacs nach einem Dateinamen.

Wenn man dem Text einen neuen Dateinamen geben will, kann man den Befehl C-x C-w (für *write*) eingeben. Bei diesem Befehl fragt Emacs *immer* nach einem neuen Dateinamen.

² „M“ steht ursprünglich für „Meta“, eine spezielle Taste auf älteren Workstation-Tastaturen.

Wer einen schon gespeicherten Text wieder mit Emacs bearbeiten möchte, kann ihn mit `C-x C-f` laden. Auch hier muss man natürlich den Dateinamen eingeben.

Mit Emacs kann man auch den Text einer kompletten Datei an der Cursorposition einfügen. Das geht mit dem Befehl `C-x i` („insert“).

6 Schnellere Cursorbewegungen

Wenn man einen längeren Text eingibt, möchte man gerne schneller im Text herumspringen können, als mit einem Zeichen pro Tastendruck. Die wichtigsten „schnelleren“ Cursorbefehle sind:

`C-a` Springt an den Anfang der aktuellen Zeile.

`C-e` Springt an das Ende der aktuellen Zeile.

`M-Pfeil links` Springt ein Wort zurück.

`M-Pfeil rechts` Springt ein Wort vorwärts.

`M-<` Springt an den Anfang des Textes.

`M->` Springt an das Ende des Textes.

`Prev` (**über den Pfeiltasten**). Springt eine Seite zurück.

`Next` (**über den Pfeiltasten**). Springt eine Seite vorwärts.

7 Textfenster

Emacs kann mehrere Dateien gleichzeitig anzeigen. Dazu gibt es ein eigenes, Emacs-internes Fenstersystem, das nichts mit den Fenstern der grafischen Benutzeroberfläche (z. B. X-Window) zu tun hat. Die Textfenster bilden Teile des Emacs-Fensters, liegen vertikal oder horizontal nebeneinander und dürfen sich nicht überlappen. Dieses System hat gegenüber „echten“ Fenstern den Vorteil, dass es genauso auf der Textkonsole funktioniert.

Jedes Textfenster hat am unteren Rand eine *Statuszeile*, die weiß auf schwarz dargestellt wird, und die einige Informationen über den Text im Fenster angibt und etwa wie folgt aussieht:

```
--*-Emacs: emacs.tex          (LaTeX Fill)--36%-----
```

Und hier die Bedeutung der einzelnen Elemente:

- Die beiden Sternchen am Anfang sagen, dass der Text verändert wurde, seitdem er das letzte Mal abgespeichert wurde.
- Dahinter steht der Name des Editors und dahinter der Name der gerade edierten Datei (ohne den Pfad), hier ist es „Emacs.tex“.
- In Klammern folgt der Editor-Modus, von dem abhängt, welche textsortenabhängigen Spezialbefehle zur Verfügung stehen (LaTeX-Fill ist ein Modus für Texte mit Druckformatanweisungen). Der Editor-Modus wird, abhängig von der Endung des Dateinamens, automatisch ausgewählt (für Dateien mit der Endung „.tex“ wird der LaTeX-Fill-Modus gewählt).
- Die Prozentzahl ganz rechts gibt an, welcher Ausschnitt der Textdatei gerade in dem Fenster angezeigt wird.

Die unterste Zeile des Emacs-Fensters ist der sogenannte *Minibuffer*. Er ist für zusätzliche Meldungen (wie etwa „Text geladen“, „Text seit letzter Speicherung nicht geändert“) und für die Eingabe von Suchbegriffen, Dateinamen und Befehlen vorgesehen.

Wer mehrere Textfenster gleichzeitig haben möchte, kann mit dem Befehl `C-x 2` das gerade aktive Textfenster (also das, worin der Cursor gerade zu sehen ist) in zwei übereinanderliegende Fenster gleicher Größe teilen. Mit `C-x 3` wird das Fenster in zwei *nebeneinander* liegende Fenster geteilt. Mit dem Befehl `C-x 0` kann man das aktuelle Fenster mit seinem „Zwillingsfenster“ wieder vereinen.

Der Befehl `C-x o` wechselt in das nächste Fenster, also rechts bzw. unterhalb des aktiven Fensters, und mit `C-x 1` wird das aktive Fenster wieder zum einzigen auf dem Bildschirm, d.h. alle anderen werden geschlossen.

8 Markierung und Region

8.1 Die Markierung

Emacs merkt sich für jeden Text, der ediert wird, zwei besondere Positionen: die Cursorposition (engl. *point*) und die Markierung (engl. *mark*). Die Cursorposition wird durch das schwarze Rechteck im Text angezeigt, die Markierung ist leider unsichtbar. Man kann sie setzen indem man den Cursor dorthin steuert, wo die Markierung gesetzt werden soll, und an dieser Stelle die Ctrl-Taste hält und die Leertaste drückt. Dann erscheint in der untersten Zeile des Emacs-Fensters die Meldung „Mark set“.

Man kann die Markierung gut verwenden, wenn man in einem Text zwei weit auseinanderliegende Textstellen hat, die man abwechselnd edieren möchte. Dann kann man eine Textstelle markieren, an die andere den Cursor setzen und mit `C-x C-x` zwischen den beiden hin- und herspringen. Mit diesem Befehl wird zur Markierung gesprungen und die bisherige Cursorposition ist die neue Markierung.

8.2 Die Region

Etliche Emacs-Befehle beziehen sich nicht nur auf ein Zeichen oder eine Zeile, sondern auf einen beliebig langen Textausschnitt. Dieser Textausschnitt heißt Region. Die Region ist immer der Textausschnitt zwischen Markierung und Cursorposition, dabei ist es egal, ob die Markierung vor oder hinter der Cursorposition steht.

9 Buffer

Jede Textdatei, die mit Emacs bearbeitet wird, wird vollständig in den Speicher geladen. Es gibt dann sozusagen zwei Versionen des Textes: einmal die Textdatei im Dateisystem, und dann die Emacs-interne Kopie, der sogenannte Buffer. Da mit den Edierbefehlen nur der Buffer-Inhalt verändert wird, ist es nötig, nach dem Edieren den Buffer in die Datei zurückzuschreiben, weil sonst alle Änderungen verlorengehen. Das macht man mit dem Befehl `C-x C-s`.

Emacs kann gleichzeitig mehrere Buffer in seinem Speicher halten, einige davon können in den Textfenstern angezeigt werden, andere sind vielleicht völlig versteckt. Um einen versteckten Buffer in ein Textfenster zu holen, gibt es den Befehl `C-x b`. Man wird nach dem Namen des Buffers gefragt, der angezeigt werden soll (das ist in der Regel der Dateiname) und tippt ihn ein³ (mit Return abschließen). Dann wird der Inhalt des Buffers angezeigt. Wenn man nicht mehr genau weiß, wie der Name des Buffers war, kann man mit `C-x C-b` eine Fenster mit der Liste

³Hier, wie auch an vielen anderen Stellen (z. B. bei der Eingabe von Dateinamen), funktioniert wie in der BASH die „tab-completion“, also das automatische Vervollständigen eines teilweise eingegebenen Namens mit der Tabulatortaste

der Emacs-Buffer laden, mit dem Cursor in die Zeile mit dem gewünschte Buffer gehen und die Leertaste drücken – damit wird der ausgewählte Buffer angezeigt.

Wenn ein neuer Buffer in einem Fenster ausgewählt wurde, wird der alte Buffer in diesem Fenster nicht gelöscht, sondern nur versteckt. Mit dem Befehl `C-x k` kann der neue Buffer gelöscht werden, und der alte wird wieder angezeigt. Das funktioniert auch mehrere Male hintereinander, falls man öfters einen neuen Buffer in das gleiche Fenster geladen hat.

10 Der Ringbuffer

Eine spezielle Rolle spielt der Ringbuffer. Er kann nicht explizit angeschaut werden, sondern nur mit Emacs-Befehlen mit Inhalt aus einem anderen Buffer geladen werden, bzw. sein Inhalt kann in einen anderen Buffer eingefügt werden. Der Ringbuffer dient praktisch als Zwischenablage, um z. B. Textteile zwischen zwei Buffern oder auch innerhalb eines Buffers zu kopieren.

Mit dem Befehl `C-k` (für *kill*) wird die aktuelle Zeile von der Cursorposition bis zum Ende gelöscht und gleichzeitig in den Ringbuffer abgelegt. Ist der Cursor schon am Ende der Zeile, dann wird das „Zeilentrennzeichen“ am Ende der Zeile gelöscht. Das bedeutet, dass die nächste Zeile an die aktuelle Zeile „angeklebt“ wird. Wenn man mehrfach hintereinander `C-k` drückt, werden also mehrere folgende Zeilen gelöscht. Jedesmal wird der gelöschte Text (oder das gelöschte Zeilentrennzeichen) an das Ende des Ringbuffers angefügt. Deshalb steht im Ringbuffer dann der gesamte gelöschte Text.

Man kann auch eine ganze Region in den Ringbuffer kopieren. Das geht, wenn die Region bereits definiert ist, mit dem Befehl `M-w`. Verwendet man stattdessen den Befehl `C-w` (für *wipe*, auswischen), dann wird die Region gelöscht, nachdem sie in den Ringbuffer kopiert wurde.

Durch den Befehl `C-y` (für *yank*, hineinziehen) wird der Inhalt des Ringbuffers an der Cursorposition des gerade bearbeiteten Buffers eingefügt. Der Inhalt des Ringbuffers wird dabei nicht gelöscht. Man kann also diesen Befehl mehrfach hintereinander ausführen und so mehrere Kopien ein- und desselben Textteils erzeugen.

Da es im Emacs nur einen Ringbuffer für alle Buffer gibt, kann man auch zwischendurch den Buffer wechseln und so Textteile von einem Buffer zum anderen kopieren.

11 Verschieben und Kopieren

Wenn man einen Textteil verschieben möchte, kann man das Konzept der Region und des Ringbuffers verwenden. Man geht dazu folgendermaßen vor:

1. Gehe an den Anfang des zu verschiebenden Textteils und markiere die Stelle mit **C-Leertaste**.
2. Gehe an das Ende des zu verschiebenden Textteils. Kopiere den Textteil mit `C-w` in den Ringbuffer. Gleichzeitig wird der Textteil im Buffer gelöscht.
3. Gehe an die Stelle, wo der Textteil eingefügt werden soll und füge ihn mit `C-y` ein.

Wer einen Textteil kopieren will, muss nur ein klein wenig anders vorgehen:

1. Gehe an den Anfang des zu kopierenden Textteils und markiere die Stelle mit **C-Leertaste**.
2. Gehe an das Ende des zu kopierenden Textteils. **Kopiere den Textteil mit `M-w` in den Ringbuffer.**
3. Gehe an die Stelle, wohin der Textteil kopiert werden soll und füge ihn mit `C-y` ein.

12 Suchen und Ersetzen

Um bestimmte Textteile zu suchen und evtl. zu ersetzen, bietet Emacs einige Befehle an.

Derjenige, der am häufigsten verwendet wird, ist **C-s**, der *inkrementelle Suchbefehl*. *Inkrementell* bedeutet, dass die Zeichenkette, nach der gesucht werden soll, nicht im ganzen einzugeben ist, sondern dass sie Buchstabe für Buchstabe eingegeben werden kann und dabei schon die Suche läuft. Immer wenn ein neuer Buchstabe eingegeben worden ist, wird das nächste Auftreten des Suchbegriffs im Text angezeigt (von der Cursorposition ab). Der Suchbegriff wird im Minibuffer eingegeben, der Cursor erscheint aber im Text an der Stelle, wo die Zeichenkette gefunden wurde. Mit der Backspace-Taste kann der zuletzt eingegebene Buchstabe wieder aus dem Suchbegriff gelöscht werden. Wird im inkrementellen Suchmodus noch einmal **C-s** eingegeben, dann springt der Cursor an die *nächste* Stelle, wo der Suchbegriff wieder auftaucht. Hat man die gewünschte Stelle gefunden, kann man mit **Return** den Suchmodus beenden und an dieser Stelle weiter edieren.

Der Befehl **C-r** funktioniert fast genauso wie **C-s**, er sucht aber *rückwärts* von der Cursorposition aus und nicht vorwärts.

Mit dem Befehl **M-%** kann man einen Begriff ersetzen: Man wird zunächst nach dem Suchbegriff und dem Ersatzbegriff gefragt. Dann sucht der Editor von der Cursorposition aus nach dem Suchbegriff und fragt an jeder Stelle, wo er ihn findet, ob er ihn dort ersetzen soll (**y**), nicht ersetzen soll (**n**), an allen weiteren Stellen ersetzen soll (**!**), oder ob der Befehl abgebrochen werden soll (**q**).

13 Wiederholen von Kommandos

Ein Befehl kann mehrere Male hintereinander ausgeführt werden, indem man vorher **C-u** und eine Zahl eintippt. Der Befehl, der danach eingegeben wird, wird so oft ausgeführt, wie die Zahl angibt. Die Eingabe **C-u 1000 M** etwa fügt tausend **M**'s ein, das Kommando **C-u 300 C-k** führt 300 mal den *kill*-Befehl aus.

14 Emacs und die Maus

Einige Befehle können bei Emacs auch über die Maus gesteuert werden:

- Wenn man auf die linke Maustaste drückt, wird der Cursor auf die Stelle gesetzt, wo gerade der Mauszeiger steht.
- Wenn man auf die rechte Maustaste drückt, wird ebenfalls der Cursor auf die Stelle gesetzt, wo gerade der Mauszeiger steht, die Markierung wird aber auf die Stelle gesetzt, wo sich der Cursor vor dem Drücken der rechten Maustaste befand. In vielen Modi wird jedoch die Funktion der rechten Maustaste umdefiniert. Im Latex-Mode z. B. erscheint beim Drücken der rechten Maustaste ein Kontextmenü.
- Mit der linken Maustaste kann man auch Textteile in den Ringbuffer kopieren: man drückt die Maustaste am Anfang des zu kopierenden Textteiles, wandert zum Ende des Textteiles, und drückt dann die rechte Maustaste. Wenn der Textausschnitt zusätzlich noch gelöscht werden soll, muss man die rechte Maustaste zweimal drücken.
- Mit der mittleren Maustaste kann man den Ringbuffer an die Stelle einfügen, wo gerade der Mauszeiger steht.
- Der Rollbalken, der an der rechten Seite jedes Textfensters vorhanden ist, zeigt an, welcher Ausschnitt des gesamten Textes gerade angezeigt wird. Er besteht aus einem weißen

Rechteck, in dem ein schwarzes Rechteck die Größe und Lage des gerade dargestellten Textes wiedergibt. Stellt man den Mauszeiger auf diesen Rollbalken, dann kann man diesen Ausschnitt verändern:

- Wenn mehrere Textfenster übereinander dargestellt werden, kann man die dazwischenliegende Statuszeile mit der Maus „greifen“ (linke Maustaste drücken) und nach oben und unten verschieben.
- Ganz oben im Emacs-Fenster findet man eine *Menüleiste*. In ihr stehen, abhängig vom Emacs-Modus, solche Namen wie „Buffers“, „File“, „Edit“ und „Help“. Wenn man mit dem Mauszeiger auf einen dieser *Menünamen* geht und die linke Maustaste drückt und gedrückt hält, bekommt man eine Sammlung von Befehlen, die man aktivieren kann, indem man mit dem Mauszeiger auf den gewünschten Befehl geht und dann die Maustaste loslässt. Das „Buffers“-Menü enthält eine Liste aller Buffer. Hier kann man denjenigen auswählen, den man sich anzeigen lassen möchte.
- Zwischen Menüleiste und oberstem Textfenster findet man zusätzlich noch eine Symbolleiste mit den wichtigsten Befehlen (z. B. „Open“, „Save“, „Copy“). Diese kann man aktivieren, indem man sie mit der linken Maustaste anklickt.

15 Emacs-Befehle direkt eingeben

Emacs kennt so viele verschiedene Befehle, dass nicht für jeden ein Tastaturkürzel benutzt werden kann. Jeder Befehl hat aber einen Namen, wie etwa *yank* für C-y oder *other-window* für C-x C-o. Diese Namen kann man direkt eintippen, wenn man vorher das Tastaturkürzel M-x eingibt. Der Name erscheint dann in der untersten Zeile des Emacs-Fensters. Wenn der Name vollständig eingegeben wurde, und die Return-Taste gedrückt wird, dann wird der Befehl ausgeführt.

16 Completions

Weil die einzelnen Datei-, Buffer- oder Befehlsnamen recht lang sein können, gibt es bei Emacs eine Methode, die Eingabe dieses Namens abzukürzen: Man gibt (z. B. nach dem Befehl C-x C-f) den Anfang des Namens ein und drückt dann die Leertaste. Entweder gibt es nur einen Namen in der Datei-, Buffer- oder Befehlsliste, die den eingegebenen Anfang hat. Dann wird der Name vervollständigt (*completed*), und man braucht nur noch die Return-Taste zu drücken. Wenn es mehrere Namen mit diesem Anfang gibt, wird die Liste dieser Namen ausgegeben, und man kann dann einen weiteren Buchstaben tippen und es mit dieser Eingabe noch einmal probieren.

17 Makros

Mit Makros kann man eine Folge von mehreren Tastatureingaben speichern und auf einen Tastendruck hin wieder ausführen. Wenn man z. B. häufig eine Zeile in Klammern setzen will, also aus

```
Dies ist eine Zeile.
```

```
eine Zeile wie
```

```
(Dies ist eine Zeile.)
```

machen will, kann man die dafür nötigen Tastatureingaben an einer Zeile ausprobieren, sie gleichzeitig aufzeichnen, und sie dann für andere Zeilen automatisch ausführen lassen. Mit dem Tastendruck `C-x (` wird die Aufzeichnung der Eingaben gestartet, mit `C-x)` wird sie abgeschlossen. Das aufgezeichnete Makro kann mit `C-x e` (für *execute*) wieder ausgeführt werden.

Für das obige Beispiel würde man folgende Eingaben machen:

`C-x (` Starte die Aufzeichnung des Makros.

`C-a` Springe zum Anfang der Zeile.

`(` Füge die öffnende Klammer ein.

`C-e` Springe zum Ende der Zeile.

`)` Füge die schließende Klammer ein.

`C-x)` Beende die Aufzeichnung des Makros.

In jeder Zeile, die einzuklammern ist, kann das nun mit `C-x e` auf einen Schlag gemacht werden.

18 Häufig benutzte Emacs-Befehle

18.1 Buffer-Befehle

`C-x C-f` Lade eine Datei in das aktive Emacs-Fenster
`C-x C-s` Speichere den gerade edierten Buffer
`C-x C-w` Speichere den Buffer unter einem neuen Namen
`C-x i` Füge eine Datei an der Cursorposition ein
`C-x k` Lösche den gerade edierten Buffer

18.2 Bewegen des Cursors

`M-Pfeiltaste` Bewege den Cursor um ein Wort
`C-a` Springe zum Anfang der Zeile
`C-e` Springe zum Ende der Zeile
`M-<` Springe zum Anfang des Buffers
`M->` Springe zum Ende des Buffers
`M-g` Springe zur angegebenen Zeile
`C-Leertaste` Setze die Markierung an die Cursorposition
`C-x C-x` Springe zur Markierung

18.3 Einfügen und Löschen

`Entf` Lösche das Zeichen hinter dem Cursor
`Backspace` Lösche das Zeichen vor dem Cursor
`C-k` Lösche vom Cursor bis zum Zeilenende
`C-w` Kopiere markierten Text in den Buffer und lösche Text
`M-w` Kopiere markierten Text in den Buffer
`C-y` Füge Buffer an der Cursorposition ein

18.4 Suchen und Ersetzen

C-s	Suche inkrementell vorwärts
M-x search-forward	Suche Zeichenkette
M-x replace-string	Ersetze Zeichenkette
M-x replace-regexp	Ersetze regulären Ausdruck
M-x query-replace	Ersetze Zeichenkette (mit Abfrage)
M-x query-replace-regexp	Ersetze reg. Ausdr. (mit Abfrage)

18.5 Fenster-Befehle

C-x 2	Teile das aktive Fenster horizontal
C-x 3	Teile das aktive Fenster vertikal
C-x 0	Lösche das aktive Fenster
C-x 1	Lösche alle anderen Fenster
C-x o	Wechsle in das nächste Fenster

18.6 Makros

C-x (Beginn einer Makro-Aufzeichnung
C-x)	Ende einer Makro-Aufzeichnung
C-x e	Makro-Ausführung

18.7 Vermischtes

C-g	Brich Emacs-Befehl ab
C-x u	Mache letzten Befehl rückgängig
C-x C-c	Beende Emacs
C-h	Rufe das Hilfesystem von Emacs auf
C-u n	Wiederhole den nächsten Befehl <i>n</i> -mal