

GNU Emacs Reference Card

(for version 20)

1 Emacs を起動する

GNU Emacs 20 を使い始めるには、`emacs` と入力すればよい。ファイルの名前を指定して編集するときは、下記のファイルの操作 Files を参照のこと。

2 Emacs を終了する

Emacs を中断する (X 環境では最小化)	C-z
Emacs を完全に終了する	C-x C-c

3 ファイルの操作 Files

ファイルを読み込む (read)	C-x C-f
ファイルをディスクに書き出す (save)	C-x C-s
すべてのファイルを保存する	C-x s
別のファイルの内容を現在のバッファに挿入する (insert)	C-x i
別のファイルの中身と取り替える	C-x C-v
作業領域 (バッファ) を指定したファイルに書き出す	C-x C-w
Version 管理 (RCS) のための持ち出し・登録	C-x C-q

4 説明を読む Getting Help

C-h (もしくは F1) を入力し、指示に従う。Emacs を初めて使う場合は、C-h t と入力してチュートリアル **tutorial** を読んだ方がよいが、このテキストだけでも用は足りる :)。

ヘルプウィンドウを消す	C-x l
次のヘルプウィンドウに移る	C-M-v
文字列に一致した説明を表示 (apropos) する	C-h a
一文字入力すると起動する関数を表示する	C-h c
関数についての説明	C-h f
現在のモードの特徴を表示	C-h m

5 問題からの復旧 Error Recovery

入力、あるいは実行を中断	C-g
システム障害で失ったファイルを復旧する	M-x recover-file
変更を元に戻す undo	C-x u or C- _u
現在のバッファを編集開始時の状態に戻す	M-x revert-buffer
乱れた画面を再描画	C-l

6 順次検索 Incremental Search

前方検索	C-s
後方検索	C-r
正規表現で検索	C-M-s
正規表現で逆方向検索	C-M-r
一つ前の検索と一致する文字列を選択	M-p
次に見つかる検索と一致する文字列を選択	M-n
順次検索を終了	RET
最後に入力した文字で生じた移動を取り消し	DEL
検索を中止	C-g

C-s か C-r を入力すると現在の方向で検索を続ける。検索中に C-g を入力すると最後の検索だけを中止する。

7 移動 Motion

移動の単位	後方	前方
文字	C-b	C-f
単語	M-b	M-f
行	C-p	C-n
行頭・行末	C-a	C-e
文章	M-a	M-e
段落	M-{	M-}
ページ	C-x [C-x]
S 式	C-M-b	C-M-f
関数	C-M-a	C-M-e
バッファの最初・最後	M-<	M->
次の画面に移動		C-v
前の画面に移動		M-v
左に移動		C-x <
右に移動		C-x >
現在の行を画面の中央になるように移動		C-u C-l

8 カットとデリート Killing and Deleting

カット・デリートの対象	後方	前方
文字 (カットではなくデリート)	DEL	C-d
単語	M-DEL	M-d
行 (行末まで)	M-0 C-k	C-k
文	C-x DEL	M-k
S 式	M-- C-M-k	C-M-k
選択範囲をカット		C-w
選択範囲をコピー		M-w
現在位置から指定した文字までカット	M-z	文字
最後にカット・コピーしたものをペーストする		C-y
直前のペーストを一つ前のコピーと取り替える		M-y

9 印を付ける Marking

カーソルの位置に印 (mark) を付ける	C-@ or C-SPC
印とポイント (point = 現矢印位置の直前) を交換	C-x C-x
印を引数の数の語だけ減らす	M-@
段落に印を付ける	M-h
ページに印を付ける	C-x C-p
S 式に印をつける	C-M-@
関数に印を付ける	C-M-h
バッファ全体に印を付ける	C-x h

10 対話置換 Query Replace

対話形式で文字列置換を開始	M-%
正規表現を使って置換を開始	M-x query-replace-regexp
置換開始後に有効なキー	
一つ置換えて、次に行く	SPC
一つ置換えて、そこにとどまる	,
置き換えないで次に行く	DEL
残りをすべて置き換える	!
一つ前の置換を元に戻す	^
対話置換から抜ける	RET
再編集に入る (抜けるには C-M-c)	C-r

11 複数のウィンドウ Multiple Windows

(コマンドが左右に2つ書いてある場合、右側は他のフレームに作用する)

他のウィンドウをすべて消す	C-x 1	
上下に2つのウィンドウを開ける	C-x 2	C-x 5 2
現在のウィンドウを消す	C-x 0	C-x 5 0
左右に2つのウィンドウを開ける		C-x 3
他のウィンドウの中を上下に移動する		C-M-v
もう一つのウィンドウに移動	C-x o	C-x 5 o
他のウィンドウのバッファを選択する	C-x 4 b	C-x 5 b
他のウィンドウにバッファを表示する	C-x 4 C-o	C-x 5 C-o
他のウィンドウでファイルを開く	C-x 4 f	C-x 5 f
他のウィンドウでファイルをリードオンリーで開く	C-x 4 r	C-x 5 r
他のウィンドウで Dired を開く	C-x 4 d	C-x 5 d
他のウィンドウでタグを探す	C-x 4 .	C-x 5 .
ウィンドウを上下に大きくする		C-x ^
ウィンドウの左右を狭くする		C-x {
ウィンドウの左右を広くする		C-x }

12 整形 Formatting

現在の行を段落にする (モードによる)	TAB
選択した範囲を段落にする (モードによる)	C-M-\
S 式を段落にする (モードによる)	C-M-q
選択した範囲を引数だけ段落にする	C-x TAB
現在のポイントの直後に改行を挿入	C-o
行の残り1つ下の行に移動	C-M-o
ポイントの上下の空白行を削除	C-x C-o
直前の行 (引数付きで次行) とつないで一行に	M-^
ポイントの両側のすべての空白を削除	M-\
ポイントに空白を一つ挿入	M-SPC
段落を整形	M-q
整形幅 (fill column) を設定	C-x f
各行の左端に付ける文字列 prefix を設定	C-x .
フェースを設定	M-g

13 大小文字変換 Case Change

単語を大文字にする	M-u
単語を小文字にする	M-l
単語の最初の文字を大文字にする	M-c
選択した範囲を大文字にする	C-x C-u
選択した範囲を小文字にする	C-x C-l

14 ミニバッファ The Minibuffer

ミニバッファでは次のキーが使用できる。

可能な限り補完する	TAB
一語だけ補完する	SPC
補完した上で実行する	RET
補完の候補を表示	?
直前の入力を取り戻す	M-p
何を入力するか、または初期値を取り戻す	M-n
正規表現で履歴を後方検索	M-r
正規表現で履歴を前方検索	M-s
操作を中止	C-g

C-x ESC ESC と入力すると、直前の操作を呼び出して編集することができる。F10 を入力すると、メニューバーを表示する。

15 バッファ Buffers

他のバッファを (名前で) 呼び出す	C-x b
すべてのバッファ名を表示する	C-x C-b
現在のバッファを kill する	C-x k

16 前後の入れ替え Transposing

文字を入れ替える	C-t
単語を入れ替える	M-t
行を入れ替える	C-x C-t
S 式を入れ替える	C-M-t

17 スpellチェック Spelling Check

現在の単語を調べる	M- $\$$
選択した範囲にある全ての単語を調べる	M-x ispell-region
バッファにある全ての単語を調べる	M-x ispell-buffer

18 タグ Tags

タグの定義を探す	M-.
次のタグを探す	C-u M-.
新しいタグ定義のファイルを指定	M-x visit-tags-table
タグテーブルの全てのファイルを正規表現検索	M-x tags-search
全てのファイルで対話置換を実行	M-x tags-query-replace
タグ検索・対話置換を続ける	M-,

19 シェル Shells

シェル操作を実行	M-!
シェル操作を選択範囲に対して実行	M-
シェル操作で選択範囲をフィルタにかける	C-u M-
シェル窓*shell*を開く	M-x shell

20 矩形 Rectangles

矩形をレジスタにコピーする	C-x r r
矩形をカットする	C-x r k
矩形をペーストする	C-x r y
矩形を挿入 (現在文字を右にずらす)	C-x r o
矩形領域を空白にする	C-x r c
各行に (固定) 文字列を付ける	C-x r t

21 省略 Abbrevs

全域省略を追加	C-x a g
モード限定省略を追加	C-x a l
この省略に全域定義を追加	C-x a i g
この省略にモード限定定義を追加	C-x a i l
明示的に省略を展開	C-x a e
直前の単語を動的に展開	M-/

22 正規表現 Regular Expressions

改行を除く全ての一字に一致		.(dot)
0か1つ以上の繰り返しに一致		*
1つ以上の繰り返しに一致		+
0か1つだけの繰り返しに一致		?
正規表現で使う文字 <i>c</i> をそのまま使う		\c
“or” 演算子		
グループ化		\{ ... \}
<i>n</i> 番目のグループと一致した文字列と同じもの		\n
単語の区切り		\b
単語の区切りでない		\B
一致するもの	最初が一致	最後が一致
行	^	\$
単語	\<	\>
バッファ	\'	\'
文字の種類	それらに一致	それら以外に一致
明示的に指定	[...]	[^ ...]
単語形式	\w	\W
<i>c</i> という形式の文字	\sc	\Sc

23 国際文字集合 International Character Sets

第1言語を指定	M-x set-language-environment
全ての入力方法を表示	M-x list-input-methods
入力方法のオン・オフ	C-\
次の操作の文字コードを指定	C-x RET c
全ての文字コードを表示	M-x list-coding-systems
好みの文字コードを選択	M-x prefer-coding-system

24 Info

Info を開く	C-h i
指定した関数・変数の Info を探す	C-h C-i
node の中での移動	
先に進む	SPC
前に戻る	DEL
node の先頭に戻る	. (dot)
node の間を行き来する	
次の node に進む	n
前の node に戻る	p
一つ上の階層に戻る	u
項目を名前を選ぶ	m
<i>n</i> (1-9) 番目の項目を選ぶ	n
相互参照に従う (1 で戻る)	f
最後に見た node に戻る	l
directory node に戻る	d
名前を指定して任意の node に移動	g
その他	
Info のチュートリアルを起動	h
Info を終了する	q
node を正規表現で検索	M-s

25 レジスタ Registers

選択範囲をレジスタに保存	C-x r s
レジスタの内容をバッファに挿入	C-x r i
ポイントの値 (位置) をレジスタに保存	C-x r SPC
レジスタに保存してあるポイントに移動	C-x r j

26 キーボードマクロ Keyboard Macros

キーボードマクロの登録を開始する	C-x (
キーボードマクロの登録を終了する	C-x)
最後に登録したキーボードマクロを実行する	C-x e
最後に登録したキーボードマクロに追加	C-u C-x (
最後に登録したマクロに名前を付ける	M-x name-last-kbd-macro
Lisp 定義形式でバッファに挿入	M-x insert-kbd-macro

27 Emacs Lisp の取り扱い操作

ポイントの前までの S 式を評価する	C-x C-e
現在の defun を評価する	C-M-x
選択範囲を評価する	M-x eval-region
ミニバッファから読んで評価する	M-:
標準システムディレクトリからファイルを読み込む	M-x load-library

28 簡単な環境設定

変数と face の設定 M-x customize

Emacs Lisp で全体に対して有効なキー割り当てを設定する (例):

```
(global-set-key "\C-cg" 'goto-line)
(global-set-key "\M-#" 'query-replace-regexp)
```

29 コマンドを書く

```
(defun コマンドの名前 (引数)
  "コマンドの説明" (interactive "型")
  本体)
```

例:

```
(defun this-line-to-top-of-window (line)
  "Reposition line point is on to top of window.
With ARG, put point on line ARG."
  (interactive "P")
  (recenter (if (null line)
                0
                (prefix-numeric-value line))))
```