

Movimentos básicos

h l k j caracter esq., dir., linha acima, abaixo
 b w palavra/token esquerda, direita
 ge e fim palavra/token esquerda, direita
 { } começo parágrafo anterior, posterior
 () começo sentença anterior, posterior
 O gm começo, meio da linha
 ^ \$ primeiro, último caracter da linha
 nG ngg linha *n*, padrão o último, primeiro
 n% porcentagem *n* do arquivo (*n deve existir*)
 n| coluna *n* da linha atual
 % ..acha próximo parêntese, colchete, coment., #define
 nH nL linha *n* do topo, base da janela
 M linha do meio da janela

Inserção & substituição → modo insert

i a insere antes, depois do cursor
 I A insere no início, fim da linha
 gI insere texto na primeira coluna
 o O abre nova linha abaixo, acima da linha atual
 rc substitui caracter sob cursor por *c*
 grc como *r*, mas sem afetar o layout
 R substitui caracteres começando do cursor
 gR como *R*, mas sem afetar o layout
 cm muda texto do comando de movimento *m*
 cc ou S muda linha atual
 C muda até o fim da linha
 s muda um caracter e insere
 ~ muda maiúsc./minúsc. e avança cursor
 g~m muda maiúsc./minúsc. do movimento *m*
 gum gUm ... texto do movim. *m* minúsculo, maiúsculo
 <*m* >*m* desloca esq., dir. texto do movimento *m*
 n<< n>> desloca *n* linhas esquerda, direita

Eliminação

x X apaga caracter sob, antes do cursor
 dm apaga texto do comando movimento *m*
 dd D apaga linha atual, até fim da linha
 J gJ junta linha atual com próxima, sem espaço
 :rd↔ apaga *r* linhas
 :rdx↔ apaga *r* linhas, põe registrador *x*

Modo Inserção

^Vc ^Vn . insere caracter *c* literalmente, valor decimal *n*
 ^A insere texto inserido anteriormente
 ^@ mesmo que ^A e pára insert → modo comando
 ^Rx ^R Rx .. insere conteúdo registrador *x*, literalmente
 ^N ^P completa texto antes, depois do cursor
 ^W apaga palavra antes do cursor
 ^U apaga todos caracteres inseridos linha atual
 ^D ^T desloca esquerda, direita largura 1 desloc.
 ^Kc₁c₂ ou c₁←c₂ entra dígrafo {*c*₁, *c*₂}
 ^Oc executa *c* em modo comando temporário
 ^X^E ^X^Y rola acima, abaixo
 <esc> ou ^[..... abandona edição → modo comando

Copiando

"*x* ... usa registrador *x* para próximo delete, yank, put
 :reg↔ ... mostra o conteúdo de todos os registradores
 :reg x↔ mostra conteúdo dos registradores *x*
 ym copia texto do comando de movimento *m*
 yy ou Y copia texto da linha atual no registrador
 p P cola registrador depois, antes da posição cursor
]p [p como p, P com recuo ajustado
 gp gP como p, P deixando cursor depois texto novo

Inserção Avançada

g?m faz codificação rot13 no movimento *m*
 n^A n^X +*n*, -*n* no número sob o cursor
 gqm formata linhas do mov. *m* com largura fixada
 :rce w↔ centra linhas na faixa *r* para largura *w*
 :rle i↔ alinha esq. linhas na faixa *r* com recuo *i*
 :rri w↔ . alinha dir. linhas na faixa *r* para largura *w*
 !mc↔ filtra linhas do mov. *m* através comando *c*
 n!!c↔ filtra *n* linhas através do comando *c*
 :r!c↔ ... filtra faixa de *r* linhas através do comando *c*

Modo Visual

v V ^V começa/pára destaque caract., linhas, bloco
 o troca posição do cursor com início do destaque
 gv inicia destaque na area visual anterior
 aw as ap .. seleciona uma palavra, sentença, parágrafo
 ab aB seleciona um bloco (), um bloco { }

Comandos desfazer & repetir

u U .. desfaz último cmd, restaura última linha mudada
 . ^R ... repete últimas mudanças, refaz último desfazer
 n. . repete últimas mudanças com cont. trocado por *n*
 qc qC ... grava, anexa caracteres digitados no regist. *c*
 q pára de gravar
 @c executa o conteúdo do registrador *c*
 @@ repete comando @ anterior
 :@c↔ executa registrador *c* como comando *Ex*
 :rg/p/c↔ executa comando *Ex c* na faixa *r*
 [onde padrão *p* combinar

Movimento complexo

- + linha acima/abaixo no 1º caracter não-branco
 B W palavra separada por espaço a esq., a dir.
 gE E .. fim de palavra separada por espaço a esq., a dir.
 n_ desce *n* - 1 linhas no 1º caracter não-branco
 g0 início da linha *screen*
 g^ g\$ primeiro, último caracter da linha *screen*
 gk gj linha *screen* acima, abaixo
 fc Fc ocorrência seguinte, anterior do caracter *c*
 tc Tc antes da ocorrência seguinte, anterior de *c*
 ; , repete último fFtT, na direção oposta
 [[]] começo de seção atrás, a frente
 []][..... fim de seção atrás, a frente
 [()] (,) não fechado atrás, a frente
 [{ }] {, } não fechado atrás, a frente
 [m]m começo, fim de java method atrás, a frente
 [#]# .. #if, #else, #endif não fechado atrás, a frente
 [*]* começo, fim de /* */ atrás, a frente

Busca & substituição

/s↔ ?s↔ procura a frente, atrás por *s*
 /s/o↔ ?s?o↔ ... procura *s* atrás, frente com desloc. *o*
 n ou /↔ repete última busca a frente
 N ou ?↔ repete última busca atrás
 # * procura atrás, a frente palavra sob o cursor
 g# g* o mesmo, mas busca também parcial
 gd gD definição local, global de símbolo sob cursor
 :rs/f/t/x↔ substitui *f* por *t* na faixa *r*
 [*x* : g—todas as ocorrências, c—confirma mudanças
 :rs x↔ repete substituição com novos *r* & *x*

Caracteres especiais em padrões de busca

`^ $` qualquer caracter, começo, fim de linha
`< >` início, fim de linha
`[c1-c2]` um caracter na faixa c₁..c₂
`[\^c1-c2]` um caracter fora da faixa
`\i \I` um identificador, excluindo dígitos
`\k \K` uma palavra-chave, excluindo dígitos
`\f \F` um nome de arquivo, excluindo dígitos
`\p \P` um caracter imprimível, excluindo dígitos
`\s \S` ... um espaço em branco, um espaço não-branco
`\e \t \r \b` $\langle esc \rangle$, $\langle tab \rangle$, $\langle \leftarrow \rangle$, $\langle \rightarrow \rangle$
`\= *` \+ ... combina 0..1, 0..∞, 1..∞ de átomos preced.
`\|` separa dois ramos ($\equiv or$)
`\(\)` agrupa padrões em um átomo

Deslocamentos em comandos de busca

`n ou +n` linha *n* abaixo na coluna 1
`-n` linha *n* acima na coluna 1
`e+n e-n` ... *n* caracteres a dir., esq. do fim da combin.
`s+n s-n` ... *n* caracteres a dir., esq. do começo combin.
`;sc` executa comando de busca *sc* próximo

Marcas e movimentos

`mc` marca posição atual com marca $c \in [a..Z]$
`'c 'C` vai para marca *c* no arquivo atual, *C* em qualquer arq.
`'0..9` vai para última posição
`' ' "` ... vai para posição antes salto, na última edição
`' [']` vai para começo, fim texto operado anterior
`:marks` mostra lista de marcas ativas
`:jumps` mostra lista de saltos
`n^0` vai para *n*^a posição mais velha na lista saltos
`n^I` vai para *n*^a posição mais nova na lista saltos

Mapeamento de teclas & abreviações

`:map c e` mapeia $c \mapsto e$ em modo normal & visual
`:map! c e` mapeia $c \mapsto e$ em modo & linha-cmd
`:unmap c` remove mapeamento *c*
`:mk f` escreve mapeamentos, ajustes... atuais no arquivo *f*
`:ab c e` adiciona abreviação para $c \mapsto e$
`:ab c` mostra abreviação começando com *c*
`:una c` remove abreviação *c*

Tags

`:ta t` pula para tag *t*
`:nta` pula para *n*^a tag mais nova na lista
`^T` pula para tag sob cursor, retorna da tag
`:ts t` lista tags que comb. e seleciona uma p/ pular
`:tj t` pula para tag ou seleciona uma se várias
`:tags` mostra lista de tags
`:npo` `:n^T` pula atrás de, para *n*^a tag antiga
`:tl` pula para última tag que combina
`^W}` :pt *t* prevê tag sob cursor, tag *t*
`^W]` divide janela e mostra tag sob cursor
`^Wz ou :pc` fecha janela visualização tag

Rolagem & multi-janela

`^E ^Y` rola linha acima, abaixo
`^D ^U` rola meia página acima, abaixo
`^F ^B` rola página acima, abaixo
`zt ou z` põe linha atual topo da janela
`zz ou z.` põe linha atual centro da janela
`zb ou z-` põe linha atual base da janela
`zh zl` rola um caracter a direita, esquerda
`zH zL` rola meia tela a direita, esquerda
`^Ws ou :split` divide janela em duas
`^Wn ou :new` cria nova janela vazia
`^Wo ou :on` coloca janela atua em uma tela
`^Wj ^Wk` move para janela abaixo, acima
`^Ww ^W^W` move para janela abaixo, acima (troca)

Comandos Ex (↔)

`:e f` edita arquivo *f*, a menos que modificado
`:e! f` edita arq. *f* sempre (por padrão relê atual)
`:wn :wN` grava arquivo e edita próximo, anterior
`:n :N` edita arquivo seguinte, anterior na lista
`:rw` grava faixa *r* no arquivo atual
`:rw f` grava faixa *r* no arquivo *f*
`:rw>>f` adiciona faixa *r* no arquivo *f*
`:q :q!` sai e confirma, sai e descarta mudanças
`:wq ou :x ou ZZ` grava no arquivo atual e sai
`<up> <down>` ... chama comandos começando com atual
`:r f` insere conteúdo do arquivo *f* abaixo cursor
`:r! c` insere saída comando *c* abaixo cursor
`:all` abre janela para cada arquivo na lista argtos.
`:args` mostra a lista de argumentos

Faixas Ex

`,` ; separa dois números linhas, def. para 1^a linha
`n` um número de linha *n* absoluto
`.` \$ linha atual, última linha no arquivo
`% *` arquivo todo, area visual
`'t` posição da marca *t*
`/p/ ?p?` linha seguinte, anterior onde *p* combina
`+n -n` +*n*, -*n* até número linha precedente

Miscelânea

`:sh` começa shell, executa cmdo. *c* em shell
`K` procura palavra-chave sob cursor com `man`
`:make` começa `make`, lê erros e vai ao primeiro
`:cn` `:cp` mostra erro seguinte, anterior
`:cl` `:cf` lista erros, lê erros do arquivo
`^L ^G` redesenha tela, mostra nome arquivo e posição
`g^G` mostra coluna cursor, linha e posição caracter
`ga` mostra valor ASCII do caracter sob cursor
`gf` abre arquivo cujo nome está sob cursor
`:redir>f` redireciona saída para arquivo *f*
`^@ ^K ^_ \` teclas sem uso, disponíveis para mapear