

0 commands terminal(shell)は命令(command)を受け取って、被演算子(operand)に演算を加える。オプションによってその振る舞いを調整する。

```
> ls -lat
command com options
prompt
      short      long
options      options
-l : long   --version
-a : all
-t : time

> ls
command/    dir/    file/    line_edit/
> ls -lat
total 0
drwxr-xr-x  5 bob  staff  160 11  3 14:46 .semi_lattice/
drwxr-xr-x  7 bob  staff  224 11  3 14:45 ../
drwxr-xr-x  2 bob  staff   64 11  3 14:45 line_edit/
...
> pwd [.] #print working dir
> pwd
/Users/bob/Desktop/lecture_25f/multi_scale_25f/linux_basic
> man ls
LS(1)          General Commands Manual       LS(1)
NAME
ls - list directory contents
SYNOPSIS
ls [-@ABC...HIOPRSTUWabcde...ghijklmnopqrstuvwxyz%,...]
[--color=when] [-D format] [file ...]
DESCRIPTION
For each operand that names a file of a type other than
directory, ls displays its name as well as any requested,
associated information. For each operand that names a
file of type directory, ls displays the names of files
contained within that directory, as well as any requested
associated information.

operator operand 演算子 被演算子
q(quit)で終了
```

## 1 dir commands

```
> mkdir hoge
> rmdir hoge
> full_path, absolute_path, 絶対パス
  > readlink -f linux_basic_key.pdf
  /Users/bob/Desktop/Lectures/MultiScale/multi_scale_25f/w7_bundle/linux_basic.key.pdf
  > realpath linux_basic_key.pdf
  /Users/bob/Desktop/Lectures/MultiScale/multi_scale_25f/w7_bundle/linux_basic.key.pdf
  > relative_path, 相対パス
> ls [.]
> ls ..
#above dir
> cd bin
> cd ..
> cd [~]
#tilde home
> open .
```



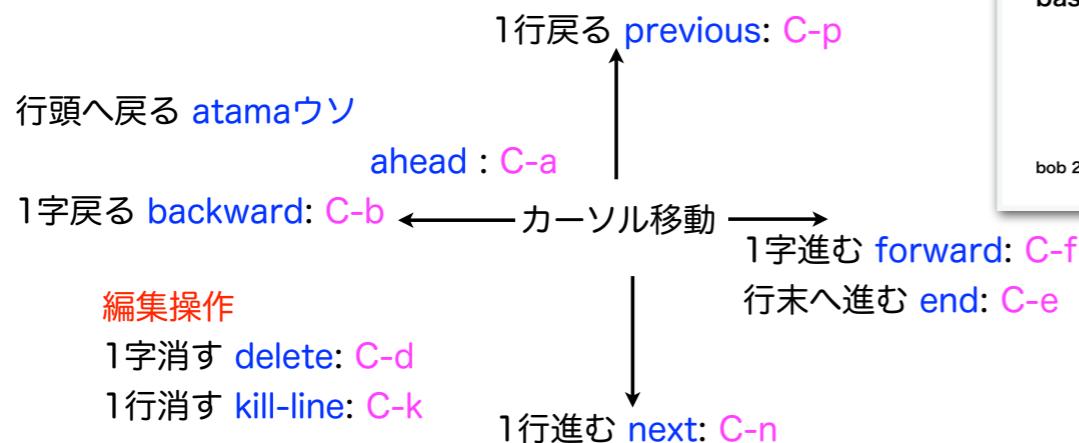
## 2 file commands

```
> mv A_file B_file
#rename A_file to B_file
> mv A_file B_dir
#move A_file to B_dir
> mv * B_dir
#move *(wild card, all files) to B_dir

> rm A_file
#remove A_file
> rm -rf B_dir
#remove recursively
and forcibly B_dir

> cp A_file B_file
#copy A_file to B_file
> cp A_file B_dir
#copy A_file in B_dir
```

## 5 line edit key bind



## linux basics

bob 2025/11/3

## 3 text commands

```
> cat A_file
#catenate A           > bat A_file
                      > sudo apt install bat

> head A_file
> tail A_file

> wc A_file
#word count A

> grep string A_file
#search string in A

> diff A_file B_file
#show diff btw A B
```

## 4 process commands

```
> ps
#process status

> fg
#fore ground
> bg
#back ground

> kill -9 pid
#kill forcibly pid process

pid : process ID
```