

使用頻度の高い UNIX コマンド

計算機応用

5E

2003.04.10

使用頻度の多い UNIX コマンドを以下に示す。本校の UNIX は IBM の AIX で、SystemV 系である。オプション等、多少異なる可能性があるが、気にせずに使ってみて欲しい。

pwd (print working directory)

【機能】

現在のワーキングディレクトリ（カレントディレクトリ）を絶対パス名で表示する。

【形式】

pwd

ls (list)

【機能】

現在のワーキングディレクトリのファイルやディレクトリーの情報を表示する。

【形式】

ls [-adFgilostux] [filename****]

【オプション】

- なし ファイル名のみ並べて出力する。
- a . (ドット) で始まる隠しファイルも含めて、すべて出力する。
- l (エル) ファイルの詳細管理情報をロング形式で出力する。
- d filename がディレクトリの時、その名前のみ表示する。
- F ファイルの種類を記号で表示する。
〈記号の意味〉
 - 表示なし プレーンなデータファイル
 - / ディレクトリファイル
 - * 実行可能ファイル
- I inode 番号を表示する。
- R ディレクトリの階層構造を表示する。
- s ファイルのサイズをブロック単位で表示する。
- t 最終更新時刻の新しいものから順に表示する。
- u l (エル) オプションとの併用時、最終更新時刻の代わりに最終アクセス時刻を表示する。
- o l (エル) オプションと同じだが、グループ名を表示しない。
- x ファイル名を横に並べて出力する。

cat (concatenate)

【機能】

ファイルの連結と表示を行う。指定した filename を順次読み取り、標準出力に出力する。この機能を利用して、複数ファイルの結合 (concatenate 鎖状につなぐ) を行うことができる。

例) cat file1 file2 > file3

上記の例では file1 と file2 を結合したものを file3 に出力している。

【形式】

cat [filename.....]

【使用上の注意】

cat file1 file2 >file1 および cat file1 file2 >file2 は、まず出力領域を確保しようとするために、読み取り前に入力データが破壊されてしまうので要注意。

cat ではデータの出力は画面単位ではない。一画面分以上の大きさのファイルを表示しようとする先頭部分はスクロールして見えなくなってしまう。その場合は more コマンドを使用する。

more

【機能】

ファイルの内容を一画面単位で出力する。

【形式】

more [filename.....]

【サブコマンド】

[space bar] 次の画面を表示する。
[return] 次の行を表示する。
q, Q more コマンドを終了する。
H サブコマンドのヘルプ情報を表示する。

cp (copy)

【機能】

filename1 を filename2 にコピーする。
directory2 配下に directory1 をサブディレクトリとして、配下のファイルごとコピーする。
各 filename (サブディレクトリ指定可) を、最後に指定した directory 配下にコピーする。

【形式】

```
cp [-ip] filename1 filename2  
cp -r[ip] directory1 directory2  
cp [-ipr] filename..... directory
```

【オプション】

-I 一先ファイルが既存の場合、置き換えを行うかどうか確認してくる。
 (置き換える場合= y 、置き換えない場合= n と入力する)
-p 内容だけでなく、最終修正時刻・アクセス許可もコピーする。
-r ディレクトリ配下のファイルごとコピーする。(ファイルがサブディレクトリ
 の場合は、その配下のファイルごとコピーする。)

mv (move)

【機能】

filename1 を filename2 に名前を変える。
directory2 が既存の場合、directory2 の配下に directory1 を移動する。
directory2 が既存でない場合、directory1 を directory2 に名前を変える。
各 filename (サブディレクトリ指定可) を、最後に指定した directory 配下に移動する。

【形式】

```
mv [-i] filename1 filename2  
mv [-i] directory1 directory2  
mv [-i] filename..... directory
```

【オプション】

-i 移動先ファイルが既存の場合、置き換えを行うかどうか確認してくる。
 (置き換える場合= y 、置き換えない場合= n と入力する)

`mkdir` (**m**ake **d**irectory)

【機能】

ディレクトリを作成する。ディレクトリ自身を表す”.” と、親ディレクトリを表す”..”の2つが、自動的に作成される。

【形式】

`mkdir` `directory`..... 指定されたディレクトリーを作成

`rmdir` (**r**emove **d**irectory)

【機能】

ディレクトリを削除する。

【形式】

`rmdir` `directory`..... 指定されたディレクトリーを削除

【使用上の注意】

削除するディレクトリは空でなければならない。ディレクトリ配下にファイルがある場合、配下のファイルごと削除するには `rm` コマンドを使う。

いずれの場合も `mkdir` コマンド同様、親ディレクトリに書き込み権が必要。

`cd` (**c**hange **d**irectory)

【機能】

`directory` で指定されたディレクトリに、ワーキングディレクトリを変更する。`directory` を指定しない場合は、ログインディレクトリに移動する。

【形式】

| | |
|----------------------------------------|-----------------|
| <code>cd</code> <code>directory</code> | 指定されたディレクトリーに移動 |
| <code>cd</code> | ログインでレク トリーに移動 |
| <code>cd</code> .. | 親ディレクトリーに移動 |

rm (remove)

【機能】

ディレクトリから各 filename の登録を削除する。
-r オプションをつけると directory 配下のファイルから順に削除していき、directory 自身も削除する。

【形式】

```
rm [-i] filename1...  
rm -r[i] directory
```

【オプション】

-i 削除していいかどうか確認のメッセージを出す。
(削除する場合=y、削除しない場合=nと入力する。)
-r directory 配下のファイルを削除し、directory 自身も削除する。
-f ファイルを強制的に削除する。

man (manual)

【機能】

指定されたコマンドのオンラインマニュアルを一画面分ずつ表示する。

【形式】

```
man [command-name]
```

【サブコマンド】

more コマンドと同じサブコマンドが使えます。

[space bar] 次の画面を表示する。
q, Q man コマンドの終了。
B 一画面前に戻る。
H サブコマンドの一覧を表示。

コマンドではないが便利な機能

↑ ↓ (history)

【機能】

前に入力したコマンドを確認したり、取り出して再実行、編集して実行する。キーを叩く回数が減り、タイプミスが減らせます。

【形式】

↑ または ↓

[ctrl]+c (ジョブの強制終了)

【機能】

ターミナルで実行したジョブを強制的に終了させる。

【形式】

[ctrl]+c ↑ [ctrl]キーとCを同時に押すとジョブが停止する。

[Tab] (補完機能)

【機能】

補完機能は、最初の何文字かを入力して[Tab]キーを押すと、自動的にディレクトリ名やファイル名を表示してくれる機能です。キーを叩く回数が減り、タイプミスが減らせます。

【形式】

たとえば、hoge hoge というファイルがあり、その中を more で見る場合、

```
more ho[Tab]
```

とすると

```
more hogehoge
```

と補完してくれます。もし、他に ho で始まるファイルがある場合は、スペルが同じところまでの表示、あるいは候補を表示します。