

情報リテラシ (3)

- Unixの基礎(2) -

シェル
コマンド

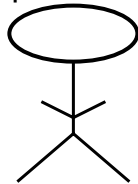
2002/4/23

情報リテラシ

1

シェル(1)

- ユーザがキーボードから入力するコマンドをコンピュータへ伝え、コンピュータが実行した結果をユーザにフィードバックするコマンド



login → シェルが自動的に起動

2002/4/23

情報リテラシ

2

シェル(2)

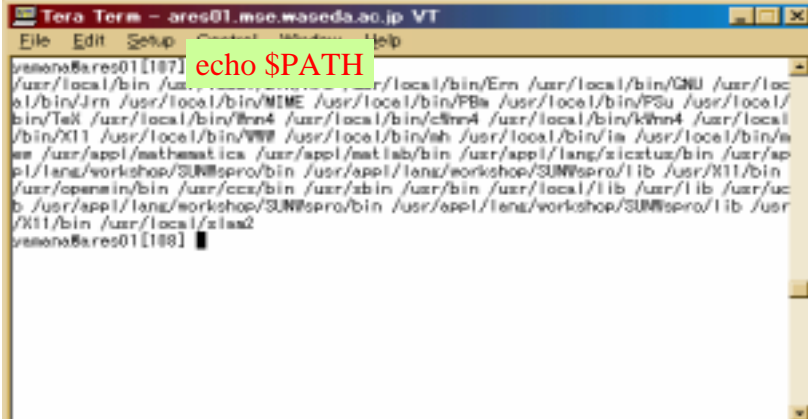
- シェルの種類
 - sh(bourne shell), ksh(korn shell), bash(bourne again shell:バッシュ) 教科書ではbash
 - csh, tcsh
- シェルは を持つ
 - echo \$PATH (パスを画面に表示する)
 - シェル上で入力されたコマンドを探すためのディレクトリ一覧

2002/4/23

情報リテラシ

3

シェル(3)



```
Tera Term - ares01.msc.waseda.ac.jp VT
File Edit Setup
ares01@ares01[187] echo $PATH
/usr/local/bin /usr/bin /usr/sbin /usr/X11/bin /usr/app/mathematica /usr/app/matlab/bin /usr/app/lang/sicstus/bin /usr/app/lang/workshop/SUNWspr/bin /usr/app/lang/workshop/SUNWspr/lib /usr/X11/bin /usr/openwin/bin /usr/ccs/bin /usr/sbin /usr/bin /usr/local/lib /usr/lib /usr/ucb /usr/app/lang/workshop/SUNWspr/bin /usr/app/lang/workshop/SUNWspr/lib /usr/X11/bin /usr/local/ziaw2
ares01@ares01[188]
```

2002/4/23

情報リテラシ

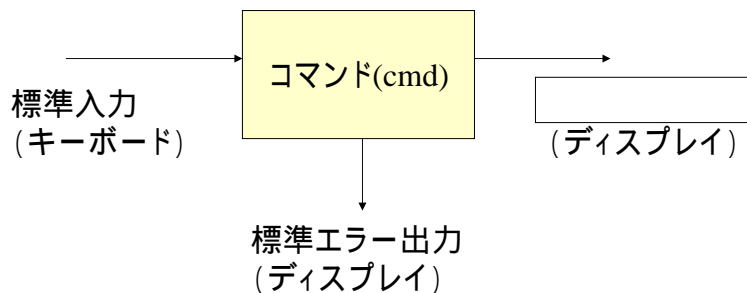
4

シェル(4)

- ワイルドカード
 - あらゆる文字列にマッチ (nullを含む)
 - ドット以外の1文字にマッチ (nullを含まず)
- リダイレクト・パイプ
 - cmd file 入力をfileから受け取る
 - cmd file 出力をfileに書き込む
 - cmd file エラー出力をfileに書き込む
 - cmd1 cmd2 cmd1の出力をcmd2の入力にする

シェル(5)

例: sort (ソート)



シェル(6)

- 直列・並列・条件実行

- cmd1 cmd2 cmd1の後cmd2を実行
- cmd1 cmd2 cmd1とcmd2を同時実行
- cmd1 cmd2 cmd1が成功した時に
cmd2を実行
- cmd1 cmd2 cmd1が失敗した時
cmd2を実行

↙
(バー)

コマンド(1)

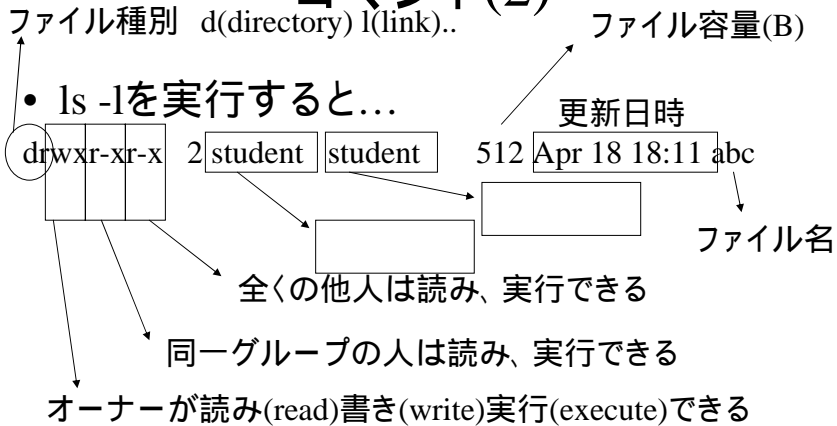
- マニュアル

- **man** cmd (わからないときにはまずmanで！)

- 基本コマンド

- pwd, cd, mkdir, rmdir, rm, ls, cat, cp, mv
- ln, du, chmod, tar, gzip, gunzip
- grep, cut, sort, uniq, find, nkf
- ps, top, kill

コマンド(2)



2002/4/23

情報リテラシ

9

コマンド(3)

- 自分の実行している には
 - kill process-number
- プロセス番号を確認するには
 -
- フォアグラウンドのプロセスを止めるには
 -
- バックグラウンドのプロセスを止めるには
 - jobsでバックグラウンドジョブを確認し kill %1等を使う。(%の後にjobsの結果表示される番号を指定する)

2002/4/23

情報リテラシ

10

シェルスクリプト(1)

- 教科書P.181に例
- たとえば、ファイル名の先頭に通し番号をつけるのは、*いちいち人間がやっていたら大変!*シェルが持つ複数のコマンド(時には外部コマンドも)を組み合わせる作業をするために新たに作ったコマンドがシェルスクリプト。

シェルスクリプト(2)

- 例:カレントディレクトリにあるファイル全ての頭に通し番号(1~)をつけるスクリプト

```
#!/bin/bash
echo "0" > /tmp/counter
counter=/tmp/counter
for file in *
do
    nums=`cat $counter`
    nums=`expr $nums + 1`
    echo $nums > $counter
    mv $file $nums$file
done
```

シェルスクリプト(3)

- 同じものをcshで書いたら...

```
#!/bin/csh
echo "0" > /tmp/counter
set counter=/tmp/counter
foreach file (*)
    set nums=`cat $counter`
    @ nums=$nums + 1
    echo $nums > $counter
    mv $file $nums$file
end
```

今日の実習

1. シェルとコマンドに慣れる