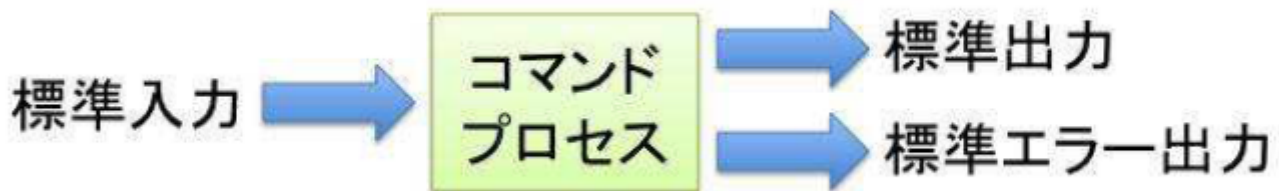


4 正規表現とパイプ

4.1 標準入出力

Linux のプログラムには、「1つの入り口と2つの出口」がある。それぞれ、標準入力・標準出力・標準エラー出力という。



- 標準入力
 - プログラムに入ってくるデータ.
 - 標準入力先は一般にキーボード.
- 標準出力
 - プログラムの実行結果を書き出す先.
 - 出力先は一般にプログラムを実行した端末のディスプレイ.
- 標準エラー出力
 - エラーメッセージを書き出す先.
 - 一般にプログラムを実行した端末のディスプレイ.

コマンドを実行した結果が画面上に表示されることを、「標準出力に出力された」と表現する。

4.2 リダイレクト

コンソールに標準出力された文字列をファイルに書き込むために使う。リダイレクトは `>` で表す。

出力のリダイレクト

例: `ls` の結果の出力のリダイレクト

```
ai@ai-VirtualBox:~$ ls > ls-output
ai@ai-VirtualBox:~$ cat ls-output
Desktop
Documents
Downloads
ls-output
Music
Pictures
Public
Templates
Videos
```

すでに同じ名前のファイルが存在しているとき、前のファイルが削除され、新しいファイルが作成される。上書きせずに追記したい場合は、アペンド `>>` を用いる (サーバの `~/.ssh/authorized_keys` にユーザの公開鍵を追加するときなど)。

cat コマンドによるファイル作成

`cat` コマンドとリダイレクトを組み合わせると、自由な内容でファイルを作成することができる。

実行例:

```
ai@ai-VirtualBox:~$ cat > cat-output
Hello.
This is cat redirect.
(ctrl + D を押す)
ai@ai-VirtualBox:~$ cat cat-output
Hello.
This is cat redirect.
```

- `ctrl + D` は、EOF (End Of File) を示すキーで、データ入力の終わりを示す。
- Linux ではデータの読み込みが最後になると、EOF が入力され終わる決まりがある。
- `cat` は EOF を受け取ったことで入力が終わったと判断し、リダイレクトを終了する。

4.3 標準エラー出力

コンピュータを操作していると、さまざまなエラーが起こることがある。

実行例:

```
ai@ai-VirtualBox:~$ ls -l tekitou > ls-l-output
ls: cannot access 'tekitou': No such file or directory
```

存在しないファイルに対して `ls` をしてそれをリダイレクトしようとしたとき、エラーメッセージが標準出力される。このエラーメッセージもリダイレクトを用いると任意のファイルにファイル出力できる。

実行例:

```
ai@ai-VirtualBox:~$ ls -l tekitou 2> ls-l-output
ai@ai-VirtualBox:~$ cat ls-l-output
ls: cannot access 'tekitou': No such file or directory
```

`2` は標準エラー出力を示す。標準出力は `1` を指定する。標準出力と標準エラー出力を1つのファイルに出力したい場合、次のように入力する。

```
ai@ai-VirtualBox:~$ ls -l tekitou > ls-l-output-second 2>&1
ai@ai-VirtualBox:~$ cat ls-l-output-second
ls: cannot access 'tekitou': No such file or directory
```

4.4 パイプ

標準入力からデータを入れることで特別な処理ができるコマンドがある。

実行例:

```
ai@ai-VirtualBox:~$ ls -l /usr/bin
```

このコマンドを実行すると画面にファイル一覧が流れるが、`/usr/bin` にはコマンドが1000以上あるので、そのままではすべてのコマンドを画面に表示することができない。そこで、以下のようにする。

実行例:

```
ai@ai-VirtualBox:~$ ls -l /usr/bin | less
```

`|` をパイプという。コマンドとコマンドをパイプ `|` でつなげることで、**パイプの前のコマンドを後ろのコマンドの標準入力とすることができる。**



4.3 のような標準出力と標準エラー出力を1つのファイルに出力する場合の方法も利用可能。

実行例:

```
ai@ai-VirtualBox:~$ ls -l xxx /usr/bin 2>&1 | less
```

4.5 grep コマンド

ファイルの中からデータを検索する。パイプを用いることで、標準入力から入ったデータに対して検索を行うことも可能。

書式:

```
grep [option] 検索条件 [指定ファイル]
```

- * のようなワイルドカードを利用した複数ファイルの指定も可能。
 - ワイルドカードとは、不特定の文字列を表現するために利用される記号のこと。
- 検索条件として正規表現が用いられることが多い。

正規表現

正規表現で使われる記号と意味:

記号	意味
^	行頭を表す
\$	行末を表す
.	任意の一字を意味する
*	直前文字の0回以上の繰り返しを意味する
[...]	.. 中の任意の一字を意味する
[^...]	.. の文字が含まれないことを意味する
\	正規表現の記号をエスケープする

正規表現の利用例:

記号	意味
^a	a で始まっている行
b\$	b で終わっている行
a.b	a と b の間に1文字入っている
[ab]ab	a もしくは b に続く ab (aab, bab)

記号	意味
[^ab]ab	a もしくは b で始まらず, ab が続くもの (e.g. xab, zab)

grep コマンドの実行例:

```
ai@ai-VirtualBox:~$ cat /etc/ssh/sshd_config | grep yes$
(/etc/ssh/sshd_config で文末がyesの行)
#PasswordAuthentication yes
#KerberosOrLocalPasswd yes
#KerberosTicketCleanup yes
#GSSAPICleanupCredentials yes
#GSSAPIStrictAcceptorCheck yes
UsePAM yes
#AllowAgentForwarding yes
#AllowTcpForwarding yes
X11Forwarding yes
#X11UseLocalhost yes
#PermitTTY yes
#PrintLastLog yes
#TCPKeepAlive yes
```

さまざまな条件を用いた grep コマンドの実行

大文字と小文字の区別に関する条件を指定したり, 否定の条件を指定したりできる.

option:

```
-e
文字列を検索パターンとして扱う

-i
検索パターンと入力ファイルの双方で, 英大文字と小文字の区別を行わない

-v
検索パターンとマッチしなかった行を選択する
```

実行例:

```
ai@ai-VirtualBox:/usr/lib/apt$ grep remove ./*
(カレントディレクトリのファイルで, remove という文字列を含むもの)
./apt.systemd.daily:#      2:  + command outputs      (remove -qq, remove 2>/dev/null, add -d)
./apt.systemd.daily:      # remove the stamp file and return 0. See coreutils bug:
./apt.systemd.daily:      # remove timestamps a day (or more) in the future and force re-check
./apt.systemd.daily:      debug_echo "end remove by archive size:  size=$size < $MaxSize"
./apt.systemd.daily:      debug_echo "skip remove by archive size:  $file, delta=$delta < $MinAge"
./apt.systemd.daily:      debug_echo "remove by archive size: $file, delta=$delta >= $MinAge (sec), siz
grep: ./methods: Is a directory
grep: ./planners: Is a directory
grep: ./solvers: Is a directory
```

```
ai@ai-VirtualBox:/usr/lib/apt$ grep remove ./* | grep -v echo
(カレントディレクトリのファイルで, remove という文字列を含み, echo という文字列を含まないもの)
grep: ./apt.systemd.daily:#      2:  + command outputs      (remove -qq, remove 2>/dev/null, add -d)
./apt.systemd.daily:      # remove the stamp file and return 0. See coreutils bug:
./apt.systemd.daily:      # remove timestamps a day (or more) in the future and force re-check
./methods: Is a directory
grep: ./planners: Is a directory
grep: ./solvers: Is a directory
```

標準出力にマッチさせる

grep コマンドはファイルの内容に一致させる以外に, 標準出力の結果を検索することが可能.

実行例:

```
ai@ai-VirtualBox:~$ ls | grep -e ^D
Desktop
Documents
Downloads
```