

# 情報科学演習 資料 2

## ディレクトリの階層構造

令和5年4月17日

### 目次

<b>1</b>	<b>復習 — ディレクトリ関係の用語と記号のまとめ</b>	<b>1</b>
<b>2</b>	<b>階層化したディレクトリの作成と利用</b>	<b>1</b>
2.1	階層化したディレクトリ . . . . .	1
2.1.1	練習 . . . . .	2
2.2	カレントディレクトリと親ディレクトリの操作 . . . . .	3
2.2.1	親ディレクトリと子ディレクトリ . . . . .	3
2.2.2	カレントディレクトリと親ディレクトリの指定法 — “.” と “..” . . . . .	3
2.2.3	練習 . . . . .	4
<b>3</b>	<b>ディレクトリ間のファイルの移動と複写</b>	<b>5</b>
3.1	ディレクトリ間のファイルの移動と複写 — mv, cp . . . . .	5
3.1.1	練習 . . . . .	5
<b>4</b>	<b>問題</b>	<b>6</b>

## 1 復習 — ディレクトリ関係の用語と記号のまとめ

ディレクトリ: ファイルを分類して格納するための器。関係コマンド一覧は前回資料末尾にある。

カレントディレクトリ: 現在使っているディレクトリ。

- コマンドの引数に与えたファイル名やディレクトリ名は、カレントディレクトリのファイルやディレクトリを指すことになる。
- カレントディレクトリを別の *directory* に変更するには、`cd directory` を実行する。

ホームディレクトリ: ユーザーが自分のファイルやディレクトリを格納するためのディレクトリ。

- ログイン直後のカレントディレクトリはホームディレクトリである。
- 記号 `~` で表すことがある。
- 引数無しで `cd` を実行すれば、ホームディレクトリがカレントディレクトリとなる。

## 2 階層化したディレクトリの作成と利用

### 2.1 階層化したディレクトリ

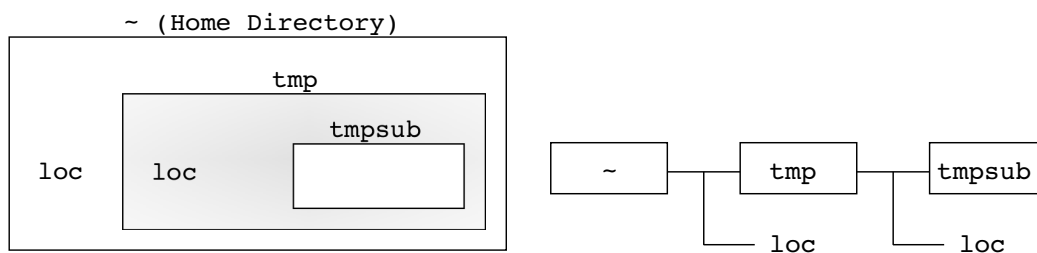


図 1: 階層化ディレクトリの例

- ディレクトリの中には、さらにディレクトリを作ることができます。この仕組みを利用すると、ファイルを大分類・中分類・小分類のように、階層的に分類して整理できます。図 1 の左の図では、ホームディレクトリの中にディレクトリ `tmp` があり、さらにその中にディレクトリ `tmpsub` がある様子を示しています。なお、`loc` は通常のファイルです。
- 右の図は、左の図を別の形式で描いたものであり、ディレクトリの中にあるファイルやディレクトリを右側に実線で結んで描画することにより、ディレクトリの階層的な構造を表しています。今後、ディレクトリの階層構造の表現には、右の図の形式のみを用います。
- 左の図における斜線は、カレントディレクトリが `tmp` であることを表現したものです。この状態でカレントディレクトリに存在するのは、斜線部に書かれたファイル `loc` とディレクトリ `tmpsub` です。これらが右の図ではどれであるか、確認してください。

コマンド操作で指定するファイル名やディレクトリ名は、カレントディレクトリのファイルやディレクトリである点に留意してください。

- 一つのディレクトリに、同じ名前のファイルやディレクトリを複数作ることはできませんが、ディレクトリが異なれば可能です。従って、図 1 のように、同名のファイル `loc` がホームディレクトリおよびディレクトリ `tmp` に存在することは許されます<sup>1</sup>。
- `ls` コマンドの引数に既存のディレクトリ `directory` を与えて

```
ls [options] directory
```

の形式で実行すれば、`directory` 内のファイル一覧が出力されます<sup>2</sup>。

もしも、カレントディレクトリにディレクトリ `tmpsub` があるならば、

```
ls tmpsub
```

は `tmpsub` に存在するファイルの一覧を出力します。

### 2.1.1 練習

図 1 に示したディレクトリとファイルをホームディレクトリ (`~`) に作ります。自分が今のディレクトリにいるのか (カレントディレクトリがどこなのか) を常に意識しながら、次の操作をしてください。

1. 通常のファイル `loc` をホームディレクトリに作ります。 `loc` の中身は `Here is my home` とします。

```
echo Here is my home > loc
```

正しくできたか、`ls` コマンドおよび `cat loc` で確認してください。

2. カレントディレクトリをホームディレクトリから `tmp` に変更します。ディレクトリ `tmp` が存在することを確認してから、

```
cd tmp
```

を実行してください。 `tmp` が存在しなければ `mkdir` コマンドで作成してから上記を実行してください。なお、今後、ディレクトリ変更の操作を、単に「`tmp` に移る・行く」のように表現することがあります。

3. `ls` を実行して、カレントディレクトリである `tmp` に存在するファイルを調べてください。 `tmp` には、まだ何も作っていませんから、何も表示されませんね。
4. `echo` コマンドを使って、ディレクトリ `tmp` の中に `loc` という通常のファイルを作ってください。ファイルの中身は `Here is tmp` とします。正しくできたか確認してください。
5. `mkdir` コマンドを使って、ディレクトリ `tmp` の中に `tmpsub` というディレクトリを作ってください。結果を確認してください。

<sup>1</sup>ファイル名 `loc` は単語 `location` からとった名前です。後の練習では、このファイルを実際に作成し、カレントディレクトリがどこであるかを確認するために使います。

<sup>2</sup>`ls` にオプション `-d` を指定すると、ディレクトリの内容ではなく、ディレクトリ自身を表示します。

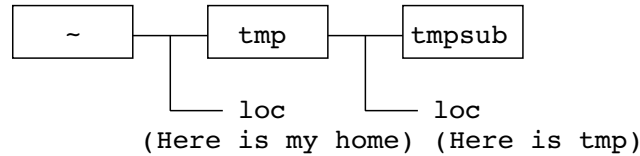


図 2: 親ディレクトリと子ディレクトリ

6. ホームディレクトリに戻るために、引数無しで `cd` を実行してください。  
ちゃんとホームに帰れましたか？ファイル `loc` の中身を確認してください。
7. `rmdir tmp` を試してください。  
ディレクトリ `tmp` を削除できませんね。 `rmdir` は中身が空のディレクトリを削除するコマンドだからです。 `tmp` は消さないでおきましょう。
8. ホームディレクトリをカレントディレクトリとしたままで、ディレクトリ `tmp` の中身 (`tmp` に存在するファイル) を確認してみましょう。 `ls` の引数にはディレクトリ名を与えることができるのでしたね。 `ls tmp` と `ls -F tmp` を実行してください。

## 2.2 カレントディレクトリと親ディレクトリの操作

### 2.2.1 親ディレクトリと子ディレクトリ

図 2 は、前節 2.1.1 で作成した、ホームディレクトリ (`~`) 内のディレクトリ/ファイルの構造を、再度示したものです。

あるディレクトリの一つ上位のディレクトリを **親ディレクトリ** (parent directory) と呼びます。図 2 では、 `tmpsub` の親ディレクトリは `tmp` であり、 `tmp` の親ディレクトリは `~` です。逆に、 `tmpsub` を `tmp` の **子ディレクトリ** (child directory) または **サブディレクトリ** (subdirectory) と呼びます。 `tmp` は `~` の子ディレクトリです。

### 2.2.2 カレントディレクトリと親ディレクトリの指定法 — “.” と “..”

カレントディレクトリの子ディレクトリは「カレントディレクトリに存在するディレクトリ」です。そのため、子ディレクトリの名前をそのままコマンドの引数として指定すれば、それがコマンドの処理対象となります。

一方、カレントディレクトリ自身や「カレントディレクトリの親ディレクトリ」は、カレントディレクトリに存在するディレクトリではありません。したがって、ディレクトリ名をコマンドの引数に与えても、コマンドの処理対象にはなりません。これらをコマンドの引数にしたければ、次の記号を使います。

- . (ドット) : カレントディレクトリ自身
- .. (ドット二つ) : カレントディレクトリの親ディレクトリ

### 2.2.3 練習

1. カレントディレクトリを、図 2 の tmpsub に変更してください。

tmpsub はカレントディレクトリに存在するディレクトリではないので、`cd tmpsub` だけではうまくいきません。まず `tmp` に移ってから、`tmpsub` に移りましょう。

2. `tmpsub` (カレントディレクトリ) に存在するファイルを `ls` コマンドで調べましょう。  
何も表示されませんね。

3. `ls` に `-F` と `-a` の二つのオプションを付けて実行してみましょう。`ls` に複数のオプションを付けるときには、それらを `-aF` のようにまとめて書くことができます。

```
ls -aF
```

`-a` は `.` で始まるものを含めて全てのファイル名を表示する `ls` のオプションでしたね。カレントディレクトリ自身とその親ディレクトリの存在を確認できましたか。

4. カレントディレクトリを、親ディレクトリである `tmp` に変更します。

まず、`cd tmp` を試しましょう。

コマンドの引数には、カレントディレクトリに存在するファイル/ディレクトリ名を指定するのでしたから、これではうまくいきません。

そこで、先ほどカレントディレクトリに存在することを確認した、親ディレクトリの記号 `..` を使います。

```
cd ..
```

エラーメッセージは出ませんね。成功です。

5. カレントディレクトリの内容を `ls -aF` で表示してみてください。`loc` というファイルが存在するはずですから、ファイルの中を見てください。

今はちゃんと `tmp` にいますね。

6. `ls -aF .` の結果を予想して、試してください。

`ls` の引数に `.` を指定すると、カレントディレクトリに存在するファイルの一覧が得られます。したがって、結果は `ls -aF` と同じです。

7. `ls -aF` や `ls -aF .` の表示では `..` も現われましたから、親ディレクトリ (ホームディレクトリ) の指定には、再度 `..` を使うことができます。

`ls` コマンドの引数に `..` を与えて、親ディレクトリに存在するファイルの一覧を表示してみましょう。

8. `..` を使ってホームディレクトリに戻りましょう。

## 3 ディレクトリ間のファイルの移動と複写

### 3.1 ディレクトリ間のファイルの移動と複写 — mv, cp

前の資料では、mv と cp の引数にファイル名を与えてファイル名の変更とファイルの複写を行いました。mv や cp の最後の引数に既存のディレクトリ名 *directory* を指定すると、mv と cp は次のとおり動作します。

`mv file directory` — *file* を *directory* に移動する

`cp file directory` — *file* を *directory* に、同じ名前のファイルとして複写する

*directory* として `.` や `..` を指定することも可能です。なお、mv の場合、*file* としてディレクトリ名を与えれば、ディレクトリを別のディレクトリに移動することもできます<sup>3</sup>。

#### 3.1.1 練習

##### 1. 準備

- (a) カレントディレクトリをホームディレクトリにするために、`cd` コマンドを引数無しで実行してください。
- (b) ホームディレクトリに通常のファイル `now` はありますか？中身はどうなっていますか？ファイル `now` が無ければ `date > now` で作成してください。
- (c) ディレクトリ `tmp` がホームディレクトリに存在することを `ls -F` で確認してください。

##### 2. ファイルの移動

- (a) ファイル `now` をディレクトリ `tmp` に移動します。`mv now tmp` を実行してください。
- (b) `ls` を実行して、ホームディレクトリから `now` が無くなったことを確認しましょう。
- (c) `ls tmp` を実行して、ディレクトリ `tmp` に `now` が存在することを確認しましょう。
- (d) カレントディレクトリを `tmp` に変更してから、ファイル `now` の内容を見てみましょう。`now` はディレクトリ `tmp` に移動できていますね。

##### 3. ファイルの複写

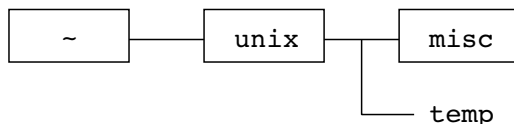
- (a) ディレクトリ `tmp` のファイル `now` を親ディレクトリ（ホームディレクトリ）に複写します。`cp now ..` を実行してください。
- (b) `ls` を実行して、カレントディレクトリ (`tmp`) に `now` が残っていることを確認しましょう。
- (c) カレントディレクトリ (`tmp`) のファイル `now` の内容を見てみましょう。
- (d) カレントディレクトリを親ディレクトリ（ホームディレクトリ）に変更して、複写した `now` の内容を見てみましょう。

---

<sup>3</sup>cp でディレクトリを複写するにはオプション `-r` が必要です。

## 4 問題

1. ホームディレクトリに存在する通常のファイルやディレクトリの名前一覧を表示しましょう。
2. その中でディレクトリはどれでしょう？ ファイルの種類を表す記号 (ディレクトリの場合は /) がつくように、ファイルやディレクトリ名の一覧を表示しましょう。
3. ホームディレクトリに存在するファイル `loc` の名前を `loc.home` に変更してください。
4. ホームディレクトリの子ディレクトリである `tmp` に存在するファイル `loc` の名前を `loc.tmp` に変更してください。
5. ディレクトリ `tmp` に存在するファイル `now` を、ディレクトリ `tmpsub` に移動してください。
6. ディレクトリ `tmp` をカレントディレクトリにした状態で、ディレクトリ `tmpsub` にファイル `now` があることを確認してください。
7. ディレクトリ `tmpsub` をカレントディレクトリにした状態で、ディレクトリ `tmp` にファイル `now` が無いことを確認してください。
8. ホームディレクトリに存在するディレクトリ `tmp` の名前を `junk` に変更してください。
9. ディレクトリ `junk` と、その中のすべてのファイルやディレクトリを削除してください。ディレクトリの削除には `rmdir` コマンドを使ってください。
10. **次回の準備** 次の図に示すように、ホームディレクトリにディレクトリ `unix` を作成し、そこにディレクトリ `misc` を作成してください。また、ホームディレクトリに存在するファイル `now` を `unix` に移動し、その名前を `temp` に変更してください。



作成したら、ホームディレクトリで `ls -R unix` を実行し、次の出力が得られることを確認してください。

```
unix:
misc temp

unix/misc:
```