

- 出席の確認のため pi にログインして待つこと
 - ちなみに演習室外からリモートログインしてもダメ

UNIX の復習

中村 匡秀

計算科学演習I 第2回講義資料

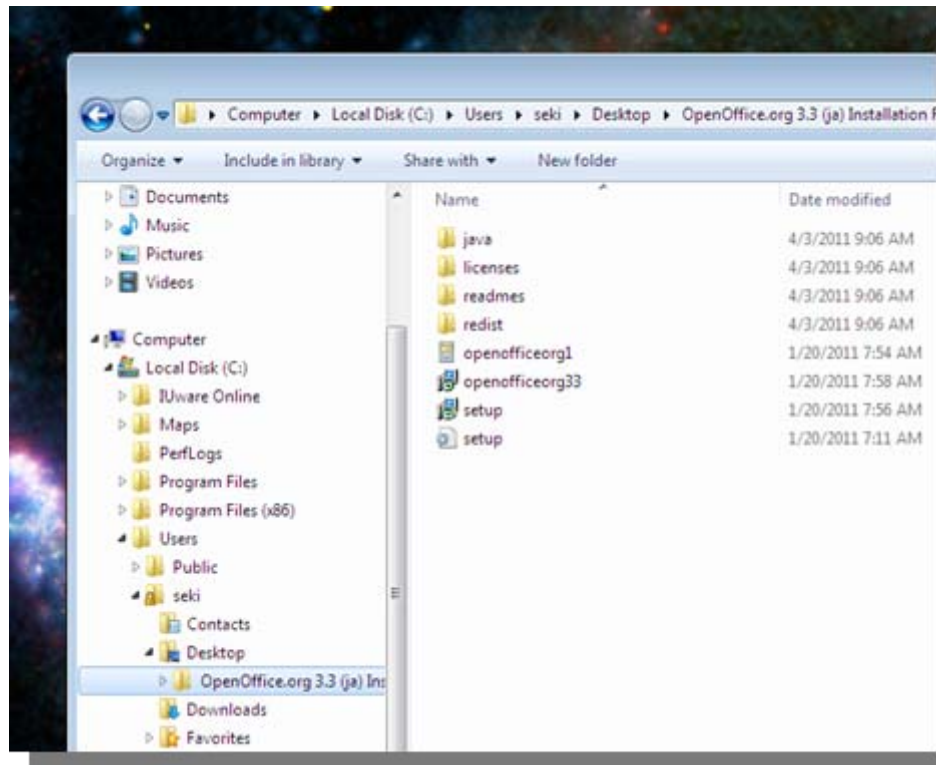
※本資料のオリジナルは関和弘先生によって作成されました。

今日やること

- UNIXの基礎
 - ディレクトリ概念
 - コマンド
 - 環境変数
 - シェルスクリプト

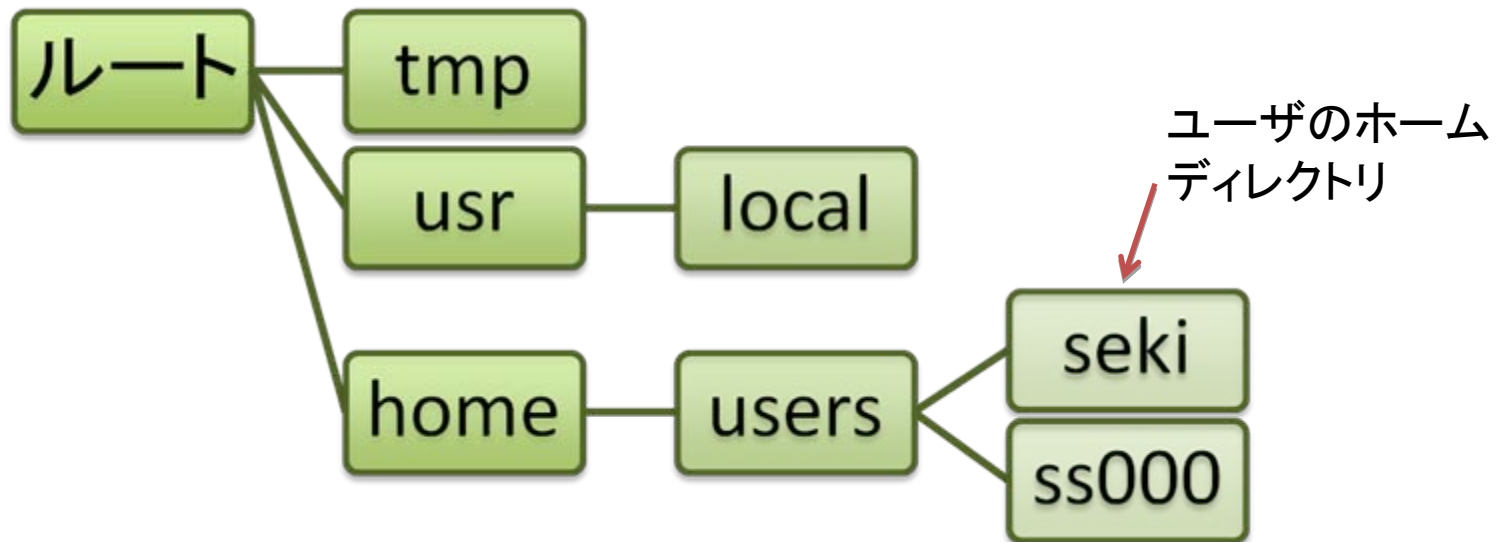
ディレクトリの概念

- Windowsのフォルダと同じ
- 木構造を持つ



ディレクトリの概念

- 木の一番上(ルート)をスラッシュ(/)で表す



ホームディレクトリ seki へのパス → /home/users/seki

UNIX コマンド

- pwd
 - 現在作業中のディレクトリ (working directory) をルートからのパスで表示
- 例
 - \$ pwd
 - /home/users/seki

UNIX コマンド

- ls
 - 指定したディレクトリにあるファイル・ディレクトリの一覧(list)を表示
- 例
 - \$ ls ← 現在のディレクトリのファイル一覧
 - \$ ls /home ← /home 下のファイル一覧
 - \$ ls /home/users ← 同様
 - \$ ls -l ← 詳細な情報を表示

UNIX コマンド

- mkdir
 - 新しいディレクトリを作成
- 例

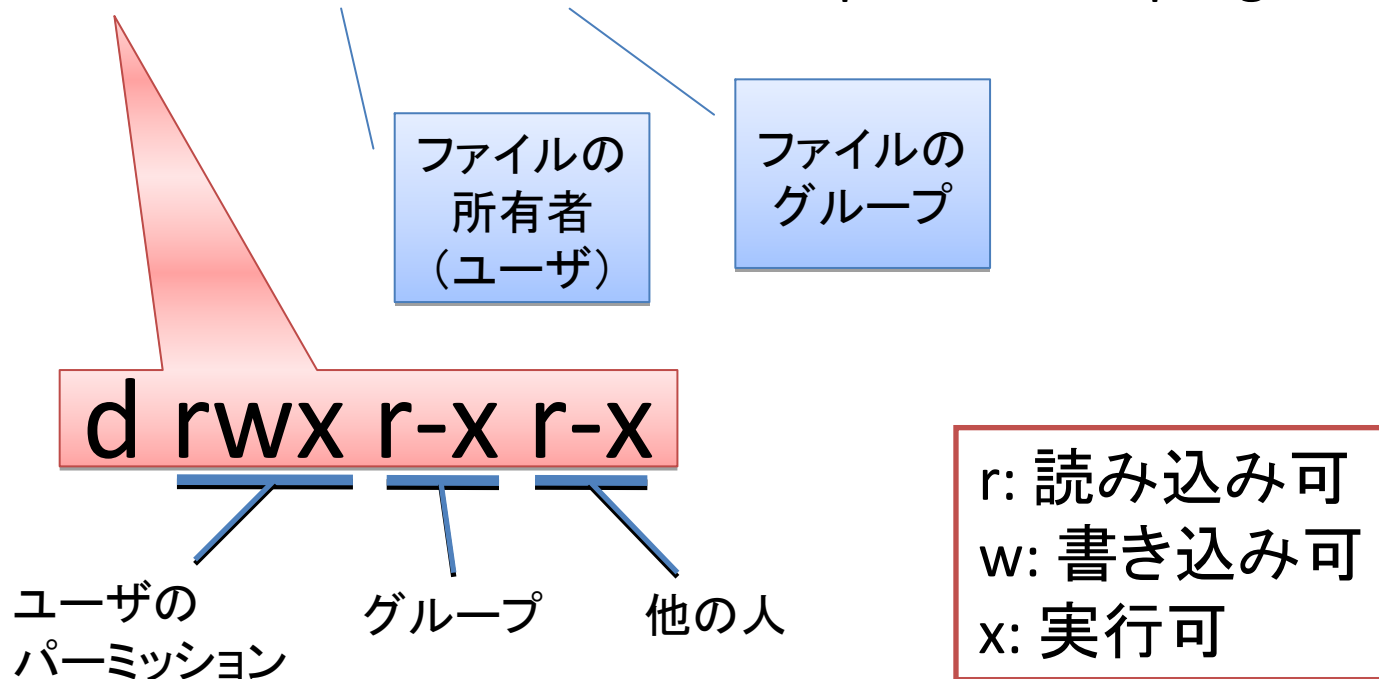
```
$ mkdir program
$ ls
program
$ ls -l
drwxr-xr-x  2 seki research 4096 Apr 20 12:02 program
```


ls の補足

- ls -l で表示される情報の意味

```
$ ls -l
```

```
drwxr-xr-x 2 seki research 4096 Apr 20 12:02 program
```



リダイレクト

- 用途

- コマンド(あるいはプログラム)の出力をファイルに保存
- コマンド(あるいはプログラム)へファイルを入力

- 例

\$ ls /home/users > files.txt ←出力を files.txt へ

\$ ls -l >> files.txt ←出力を files.txt の後ろへ追加

\$./prog < input.txt ←input.txt を prog の入力とする

UNIX コマンド

- chmod
 - ファイル・ディレクトリのパーミッションを変更

- 例

```
$ ls -l files.txt
```

```
-rw-r--r-- 1 seki research 45469 Apr 20 14:59 files.txt
```

```
$ chmod go-r files.txt
```

```
$ ls -l files.txt
```

```
-rw----- 1 seki research 45469 Apr 20 14:59 files.txt
```

グループ (**g**roup) と他人 (**o**thers) から、
読み込み (**r**ead) のパーミッションを除く (-)

chmod の補足

- 対象を指定して変更

- **u**(ser), **g**(roup), **o**(ther) に対して, 読み込み(**r**), 書き込み(**w**), 実行(**x**)権限を与えるか(**+**), 与えないか(**-**)

- 例: `ug+rw, go+r, go-rwx, ...`

- 値を直接指定して変更

- 3 bit ずつで区切って, 2進数→10進数へ変換

- 例: `rw- rw- r--` → 110 110 100 → 664 → `chmod 664 files.txt`
`rwx --- ---` → 111 000 000 → 700 → `chmod 700 files.txt`

UNIX コマンド

- cat
 - ファイルの中身を表示
- 例
 - \$ cat files.txt

UNIX コマンド

- head
 - ファイル先頭の10行を表示
- 例
 - \$ head files.txt
 - \$ head -20 files.txt ← 最初の20行を表示

UNIX コマンド

- tail
 - ファイルの最後10行を表示
- 例
 - \$ tail files.txt
 - \$ tail -20 files.txt ← 最後の20行を表示

UNIX コマンド

- less

- ファイルの中身を一画面ずつ表示

- 例

- \$ less files.txt

- 以降, スペースキーを押すごとに一画面ずつ進む.

- bで一画面戻せる.

- カーソルキーで一行ずつ動かせる.

UNIX コマンド

- cd
 - 指定したディレクトリへ移動 (change directory)

- 例

cd ← 自分のホームディレクトリへ
移動

cd program ← ホームにある program に移動

cd /etc ← ルート下の etc に移動

cd .. ← 親ディレクトリへ移動

cd ~/program ← ホームにある program に移動

cd を実行するたびに, pwd を実行してみよう

パスに関する補足

- 絶対パス

- ルートから指定

- 例

```
$ cd /home/users/seki/program
```

自分のホームは「~」と書ける.
なので, ~/program で良い.

- 相対パス

- カレントディレクトリからのパスを指定

- 例

```
$ cd program # カレントが/home/users/sekiだと仮定
```

```
$ cd seki/program # カレントが/home/usersだと仮定
```

UNIX コマンド

- mv
 - ディレクトリ(ファイル)を移動, または名前を変更

- 例

```
$ mv program fortran
```

```
$ mkdir enshu
```

```
$ mv fortran enshu
```

```
$ ls
```

```
enshu
```

```
$ ls enshu
```

```
fortran
```

シナリオ:

program を fortran にして, 新しい
ディレクトリ enshu の下に移動

二つ目の引数がディレクトリか
どうかで挙動が変わる

UNIX コマンド

- cp
 - ファイル・ディレクトリのコピー
- 例
 - \$ cp files.txt files-copy.txt
 - \$ ls
 - \$ cp enshu enshu-copy
 - cp: omitting directory `enshu`
 - \$ cp -r enshu enshu-copy
 - \$ ls

UNIX コマンド

- rm

- ファイルを削除

- 例

- \$ rm files-copy.txt

- \$ ls

- \$ rm enshu-copy

- rm: cannot remove `enshu-copy': Is a directory

UNIX コマンド

- rmdir
 - ディレクトリを削除
- 例
 - \$ rmdir enshu-copy
 - rmdir: `enshu-copy/`: Directory not empty

 - \$ rmdir enshu-copy/fortran
 - \$ rmdir enshu-copy

rm -r enshu-copy でも良い

UNIX コマンド

- grep

- 指定の文字列を含む行を出力

- 例

```
$ grep seki files.txt
```

```
seki
```

```
drwx----- 6 seki          research 4096 ...
```

パイプ

- 用途

- コマンド(プログラム)の出力を他のコマンド(プログラム)の入力とする

- 例

- ```
$ ls /home/ss | grep m13
```

- ```
m130x000
```

- ```
m130x014
```

- ```
m130x028
```

- ```
...
```



# UNIX コマンド

- man
  - コマンドのマニュアルを表示
- 例
  - \$ man ls
  - \$ man cp
  - ...

今回紹介したコマンドには様々なオプションがあります。慣れて来たらいろいろと試してみましょう！

# 環境変数

- OSが用いる変数. システムによって定義されるもの, ユーザが任意に定義するものがある
- 定義の仕方
  - \$ export VAR=value
- 変数の値の表示
  - \$ echo \$VAR
  - \$ set # すべての環境変数を表示

# シェルスクリプト

- UNIXコマンドで作ったプログラム. バッチジョブに便利

ids.sh

```
#!/bin/sh
cd ~
mkdir masters
cd masters
ls /home/ss | grep "^m13" > m1
ls /home/ss | grep "^t10" > b4
```

```
$ chmod u+x ids.sh
$./ids.sh
```

# 課題

自宅か研究室から pi にログインし, 以下を実行しなさい.

1. ディレクトリ ~/unix を作成
2. ~/unix に移動
3. emacs を起動し次を実行
  - 前回作成した ~/test.c を開き, ~/unix/test.c に保存 (C-x C-w)
  - kansou.txt を新規作成. 今日の演習の感想・コメントを**英語**で書く. 1~2文で良い.
4. test.c をコンパイルし, 実行結果を test.txt に書き込む.
5. kansou.txt のパーミッションを自分だけ読み書き可能に変更.
6. 以下を実行する.

```
$ cd ~/unix
```

```
$ cat kansou.txt test.txt > report.txt
```

```
$ ls -l kansou.txt >> report.txt
```

**←report.txt の中身を確認すること!**

```
$ cat report.txt | mail -s 自分のユーザ名 kobeuniv.compra1@gmail.com
```

**(締切: 次回授業日の前日23:59まで)**