

# UNIXコマンド基礎

# コマンドについて

本研究会において

基礎だけで良いので理解

- コマンドの使用(ls, dir, cdなど)
- Terminal/コマンドプロンプト(Linux, Mac/Windows)の使用
- Linux (参考Web: <https://linuxfan.info/ubuntu-open-terminal-emulator>)  
[Ctrl] + [Alt] + [T]
- Mac (参考Web: <https://itips.krsw.biz/where-is-terminal-in-mac-how-to-run/>)  
SpotlightにTerminalと入力
- Win (参考Web: <https://atmarkit.itmedia.co.jp/ait/articles/1806/14/news017.html>)  
[Windows] + [r]で検索を開き、cmdと入力

精通してくると・・・

- PCにファイル変更を一括指示(下図)
- カビゴンを表示(右図)

も可能(興味があれば勉強しよう)

```
[sugiharakenta@mar90-247] [09:00] [~]
-> $ find . -type f | xargs file | grep CRLF | awk
-F: '{print $1}' | xargs nkf -Lu --overwrite
```



```
Snorlax
Welcome to fish, the friendly interactive shell
Type help for instructions on how to use fish
[sugiharakenta@mar90-247] [08:24] [~]
-> $
```

# Linux, Mac

ディレクトリ構造について

例: /home/xxx/egs5(xxx: ユーザー名)

意味: 「home」の下の「xxx」の下の「egs5」

「/」は下に含まれるという解釈

「.»で現在のディレクトリ

「..」で一つ上のディレクトリ

以下、A, Bはファイル名を表すとする

コマンド	使い方	意味
ls	ls	ディレクトリ内のファイルの表示
cd	cd egs5/calc/ cd ..	egs5/calcに移動 一つ上のディレクトリに移動
mv	mv A B	ファイルAの名前をBに変更
cp	cp A B	ファイルAをBという名前でコピー
pwd	pwd	現在のディレクトリを表示

# Windows

ここでは省略するが、  
WindowsでもWSLで  
UNIXコマンド使用可能

ディレクトリ構造について

例: C:¥egs5¥calc

意味: 「Cドライブ」の下の「egs5」の下の「calc」

「¥」はLinux/Macの「/」と同じ

「.»で現在のディレクトリ

「..」で一つ上のディレクトリ

以下、A, Bはファイル名を表すとする

コマンド	使い方	意味
dir	dir	ディレクトリ内のファイルの表示
cd	cd egs5/calc/ cd ..	egs5/calcに移動 一つ上のディレクトリに移動
ren	ren A B	ファイルAの名前をBに変更
COPY	COPY A B	ファイルAをBという名前でコピー
@cd	@cd	現在のディレクトリを表示

# コマンドの例(Mac)

```
[sugiharakenta@mar90-247] [19:13] [~]
```

```
-> $ pwd
```

```
/Users/sugiharakenta
```

現在のディレクトリの表示

```
[sugiharakenta@mar90-247] [19:13] [~]
```

```
-> $ cd work/kek/workshop/egs/
```

work/kek/workshop/egsに移動

```
[sugiharakenta@mar90-247] [19:13] [~/w/k/w/egs]
```

```
-> $ pwd
```

```
/Users/sugiharakenta/work/kek/workshop/egs
```

```
[sugiharakenta@mar90-247] [19:13] [~/w/k/w/egs]
```

```
-> $ ls
```

```
commandline.pptx  egs_workshop.txt
```

ディレクトリ内のファイルの表示

```
[sugiharakenta@mar90-247] [19:13] [~/w/k/w/egs]
```

```
-> $ cp egs_workshop.txt egs_workshop1.txt
```

egs\_workshop.txtを  
egs\_workshop1.txtにコピー

```
[sugiharakenta@mar90-247] [19:13] [~/w/k/w/egs]
```

```
-> $ ls
```

```
commandline.pptx  egs_workshop.txt  egs_workshop1.txt
```

```
[sugiharakenta@mar90-247] [19:13] [~/w/k/w/egs]
```

```
-> $ mv egs_workshop.txt egs.txt
```

egs\_workshop.txtを  
egs.txtに名称変更

```
[sugiharakenta@mar90-247] [19:13] [~/w/k/w/egs]
```

```
-> $ ls
```

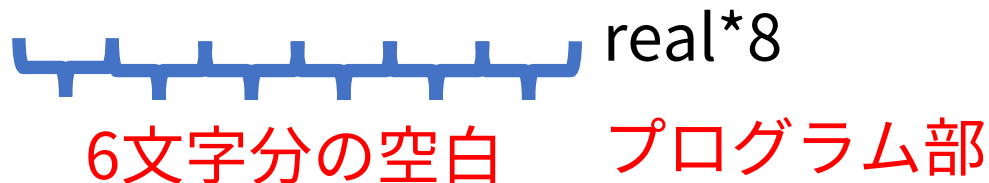
```
commandline.pptx  egs.txt  egs_workshop1.txt
```

# Fortran基礎

# Fortran77について

## Fortran77の特徴

- ・大文字・小文字の区別なし
- ・7~72カラムまでにプログラムを記述

 real\*8  
6文字分の空白      プログラム部

変数の型: 整数(型)などを明示的に定義

型	記述方法
整数	integer
実数	real
倍精度実数	real*8, double precision
文字	character

# 組み込み関数について

組み込み関数: 最初から用意されている関数(例: open, write)

関数とは: 数学における関数と同等

例:  $y = x^2 \rightarrow x = 2$ を与えると、2乗演算を加えて4を返却

open(番号, file名)  $\rightarrow$  file名を与えると、fileという入れ物の蓋を開いた状態を返却  
(番号: fileに与えるID)

マイナンバーみたいなもの



関数	使い方	演算内容
open	open(17, "myfile.txt")	myfile.txt(17番)を開く
write	write(17, *) "Hello"	myfile.txt(17番)にHelloを追記
dsqrt	dsqrt(2)	$2^{1/2} = 1.4142 \dots$ を返却