

# Linux入門

大阪大学大学院医学系研究科 遺伝統計学  
東京大学大学院医学系研究科 遺伝情報学  
理化学研究所生命医科学研究センター システム遺伝学チーム

<http://www.sg.med.osaka-u.ac.jp/index.html>

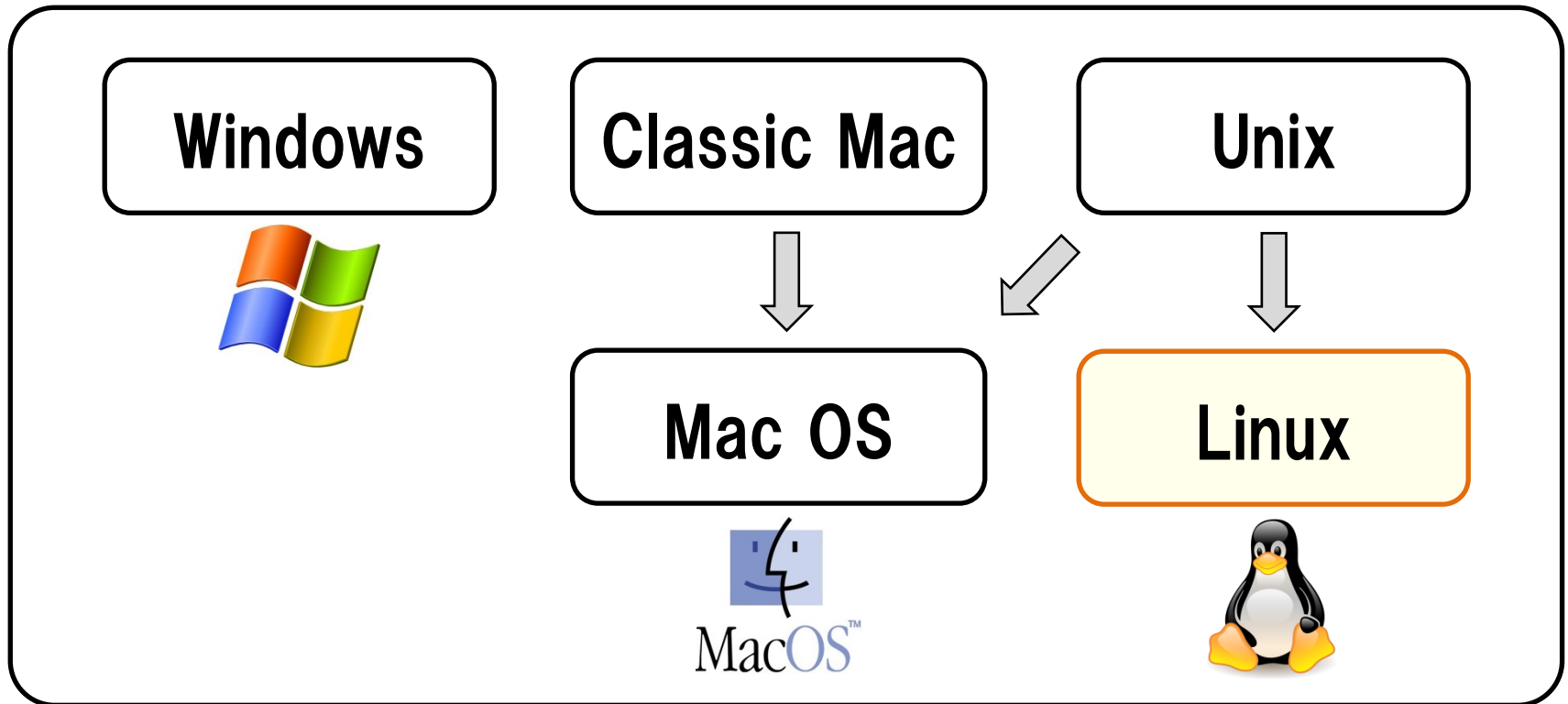
## Linux

- ① Linuxについて
- ② WSL2とCygwinについて
- ③ Linuxコマンド実習

本講義資料は、Windows PC上で  
C:\¥SummerSchool にフォルダを配置するこ  
とを想定しています。

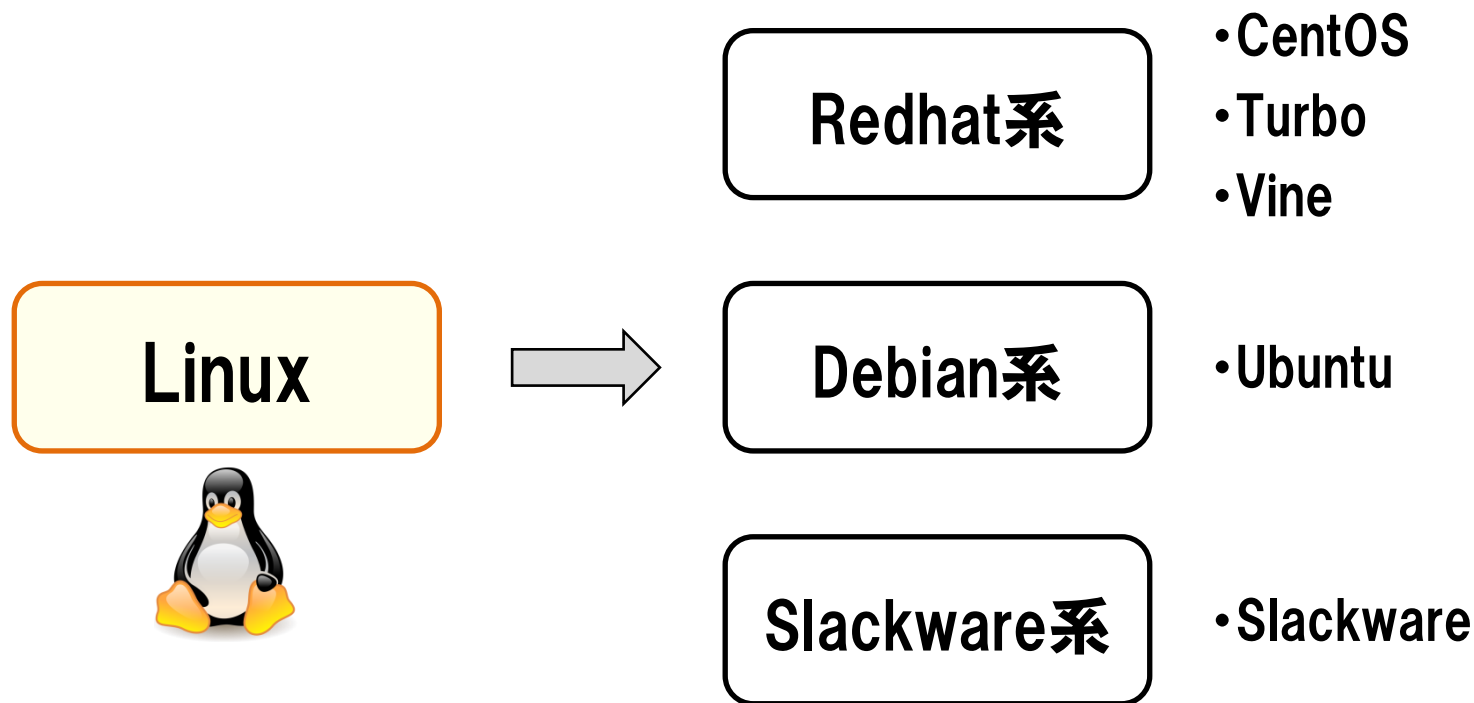
# ① Linuxについて

## OS(Operating System)



- **Linux**は、OS(Operating System)の一種です。
- OSはコンピューターを動かす基本ソフトウェアの一種で、主要なものに、**Windows**や**Mac OS**が挙げられます。
- Linuxは、**Unix**というOSと互換性を持つように作られたUNIX系OSです。

# ① Linuxについて



- **Linux**は、フリーのオープンソースなソフトウェアです。
- 多くの人が開発に参加してきた経緯から、Linuxの**ディストリビューション**（配布形式）には、**Redhat系**や**Debian系**など、様々な種類が存在します。
- ゲノム解析用の大規模計算機サーバーでは、Redhat系に属する**CentOS**が使われる例が多いようです。  
(初心者ユーザーとしては、違いをあまり気にしなくて大丈夫です。)

# ① Linuxについて

```
nickel.broadinstitute.org:22 - Tera Term VT
ファイル(E) 編集(E) 設定(S) コントロール(Q) ウィンドウ(W) ヘルプ(H)
 2 root    RT -5    0    0    0 $ 0.0 0.0 1:11.40 migration/0
 3 root    RT  34 19    0    0 $ 0.0 0.0 0:01.95 ksoftirqd/0
 4 root    RT -5    0    0    0 $ 0.0 0.0 1:40.08 migration/1
 5 root    RT  34 19    0    0 $ 0.0 0.0 0:05.49 ksoftirqd/1
 6 root    RT -5    0    0    0 $ 0.0 0.0 0:46.58 migration/2
 7 root    RT  34 19    0    0 $ 0.0 0.0 0:05.78 ksoftirqd/2
 8 root    RT  10 -5    0    0 $ 0.0 0.0 0:10.85 events/0
 9 root    RT  10 -5    0    0 $ 0.0 0.0 0:03.68 events/1
10 root    RT  10 -5    0    0 $ 0.0 0.0 0:05.34 events/2
11 root    RT  10 -5    0    0 $ 0.0 0.0 0:00.00 khelper
84 root    RT  10 -5    0    0 $ 0.0 0.0 0:00.01 kthread
90 root    RT  10 -5    0    0 $ 0.0 0.0 0:01.00 kblockd/0
91 root    RT  10 -5    0    0 $ 0.0 0.0 0:00.60 kblockd/1
92 root    RT  10 -5    0    0 $ 0.0 0.0 0:00.49 kblockd/2
93 root    RT  14 -5    0    0 $ 0.0 0.0 0:00.00 kacpid
327 root   RT  11 -5    0    0 $ 0.0 0.0 0:00.00 cqueue/0
328 root   RT  12 -5    0    0 $ 0.0 0.0 0:00.00 cqueue/1
329 root   RT  11 -5    0    0 $ 0.0 0.0 0:00.00 cqueue/2
332 root   RT  11 -5    0    0 $ 0.0 0.0 0:00.00 khubd
334 root   RT  10 -5    0    0 $ 0.0 0.0 0:00.00 kseriod
416 root   RT  15  0    0    0 $ 0.0 0.0 0:00.40 khungtaskd
417 root   RT  16  0    0    0 $ 0.0 0.0 4:35.37 pdfflush
419 root   RT  10 -5    0    0 $ 0.0 0.0 41:13.10 kswapd0
420 root   RT  12 -5    0    0 $ 0.0 0.0 0:00.00 aio/0
421 root   RT  13 -5    0    0 $ 0.0 0.0 0:00.00 aio/1
422 root   RT  12 -5    0    0 $ 0.0 0.0 0:00.00 aio/2
629 root   RT  11 -5    0    0 $ 0.0 0.0 0:00.00 kpsmoused
bash:nickel:/medpop/plenglab_scratch/yokada/ImputeHLA/IMPUTE 509 $ ls
AsianRef_YHEO_YE.bed*  AsianRef_YHEO_YE.ped*  HGDP_t36c.map*      KG_t36c.map*      olds/
AsianRef_YHEO_YE.bim*  HGDP_t36c.bed*        HGDP_t36c.ped*     KG_t36c.ped*     KG_t36c_othASN.map*
AsianRef_YHEO_YE.fam*  HGDP_t36c.bim*       Imputed_Phased/    KG_t36c_nonASN.info*  KG_t36c_othASN.ped*
AsianRef_YHEO_YE.info* HGDP_t36c.fam*       Imputed_TIDGC/     KG_t36c_nonASN.map*  Ref/
AsianRef_YHEO_YE.map*  HGDP_t36c.info*      KG_t36c.info*      KG_t36c_nonASN.ped*  Ref_Phased/
bash:nickel:/medpop/plenglab_scratch/yokada/ImputeHLA/IMPUTE 510 $
bash:nickel:/medpop/plenglab_scratch/yokada/ImputeHLA/IMPUTE 510 $
```



- Linuxの特徴は、メインのユーザーインターフェースが、コマンド入力である、という点です。
- キーボードからの文字入力ですべての操作を(主に)行います。

# ① Linuxについて

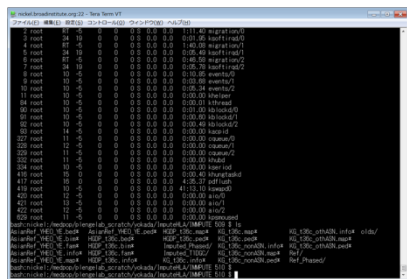


- 一方で、WindowsやMac OSでは、メインのユーザーインターフェースが、アイコン操作で動く、GUI(Graphical User Interface)形式になります。
- キーボードからの文字入力に加えて、マウスを使って操作します。

# ① Linuxについて

CUI

GUI



- GUIは直観的な操作が可能ですが、計算リソース(CPUやメモリ)を多く消費する点や、操作の記録ができない点が課題です。
- CUIは慣れるまで操作がやや難しいですが、**計算リソースの消費が少なく、入力コマンドの履歴として、操作を記録することができます。**
- GUIとCUI、どちらも重要ですが、**データ解析の分野ではCUIが便利です。**

# ① Linuxについて



CUIが難しそうで・・・



最初は難しいですが、すぐ慣れるようになります。

- CUIは、最初は難しく感じるかもしれませんが、**慣れるととても便利**です。
- コンピューターのリソースを、最大限データ解析に割り振るためにも、リソース消費の少ないCUIは役立ちます。
- データ解析の研究を行う場合、**操作が記録できることはとても重要**です。



# ① Linuxについて

クライアント端末  
(Windows/Mac)



大規模計算機  
サーバー  
(Linux)



+

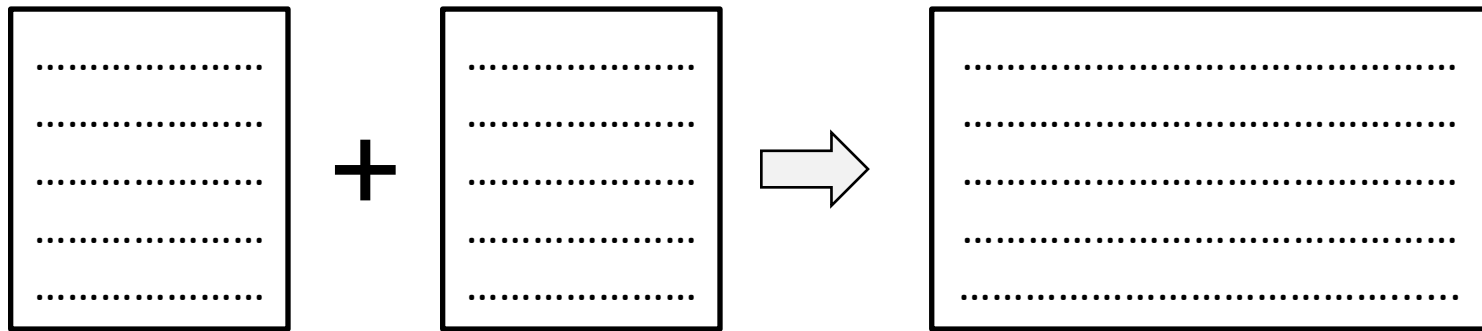
大容量データ  
ストレージ



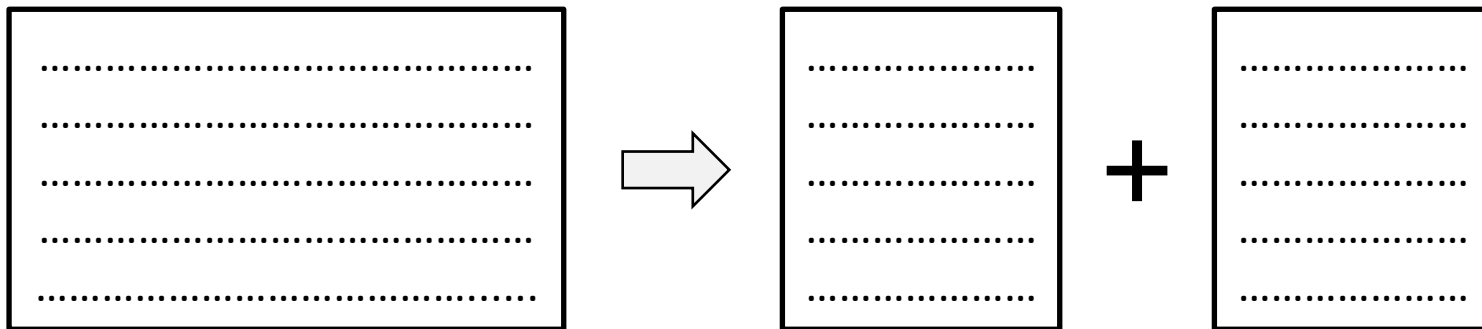
- ゲノムデータ解析の分野では、データ容量や計算リソースの規模から、個別のクライアント端末PCのみを使用した解析は難しいです。
- 大規模計算機サーバー(Linux)および大容量データストレージに、クライアント端末からログインして解析を行うのが一般的です。
- その観点からも、Linux環境に慣れ親しんでいることが重要です。

# ① Linuxについて

## テキストファイルの結合



## テキストファイルの分割



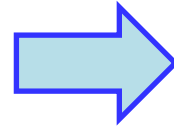
- Linuxは、**テキスト形式ファイルの取り扱い**(閲覧・整形・切り貼り・照合)に関する機能が充実しています。
- ゲノムデータ解析では、ゲノムデータや解析結果を巨大なテキストファイルとして扱う例が多く、Linuxのテキストファイル処理機能は重宝します。

## Linux

- ① Linuxについて
- ② WSL2とCygwinについて
- ③ Linuxコマンド実習

## ② WSL2とCygwinについて

クライアント端末  
(Windows/Mac)



各端末PCから  
ユーザーアカウントと  
パスワードを使って  
ログイン

大規模計算機  
サーバー  
(Linux)



- Linuxの使用ケースとして多いのは、大規模計算機サーバー(Linux)に、各自がクライアント端末(Windows/Mac)から、ネットワーク経由でログインする形です。
- **サーバーの構築は専門スキルを要する**ため、Linux環境に触れる際の参入障壁ともなっていました。

## ② WSL2とCygwinについて

Linux



Windows



- Windows上に、**擬似的にLinux環境を構築**することで、サーバーが無くてもLinux環境に触れてみるすることができます。  
(クライアント端末OSそのものをLinuxにする例もあります。また、Mac OS Xは構造上UNIXに近いので、疑似環境の構築は必ずしも必要ありません。)
- 擬似的なLinux環境は複数あり、本講義では**WSL2**と**Cygwin**を使います。

## ② WSL2とCygwinについて

# Cygwin

Get that [Linux](#) feeling - on Windows

This is the home of the Cygwin project

### What...

...is it?

Cygwin is:

- a large collection of GNU and Open Source tools which provide functionality similar to a [Linux distribution](#) on Windows.
- a DLL (cygwin1.dll) which provides substantial POSIX API functionality.

<https://www.cygwin.com/>



Welcome to the Windows Subsystem for Linux  
Update Setup Wizard

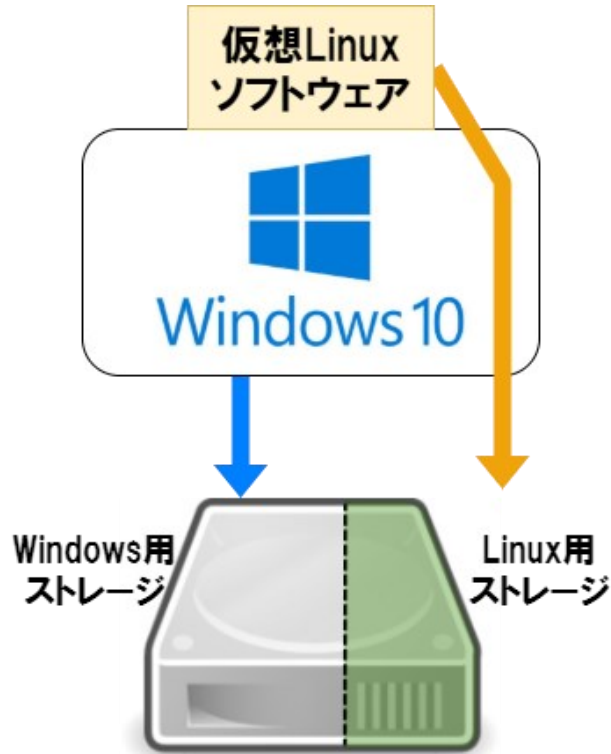
The Setup Wizard will install Windows Subsystem for Linux  
Update on your computer. Click Next to continue or Cancel to  
exit the Setup Wizard.

<https://docs.microsoft.com/ja-jp/windows/wsl/>

- **Cygwin**は、Windows上で動作する、**フリーのLinux疑似環境ソフト**です。
- **但し、あくまで”疑似”環境なので、細かい点では、本家のLinuxとは動作が異なることに注意が必要です。**
- **WSL2**(Windows Subsystem for Linux 2)は、Windows 10から導入された、**Microsoft純正の仮想Linux環境**です。**Windowsを中継せずにLinuxの命令をPCで実行でき、ほとんどの動作がLinuxと同じように行えます。**<sup>14</sup>

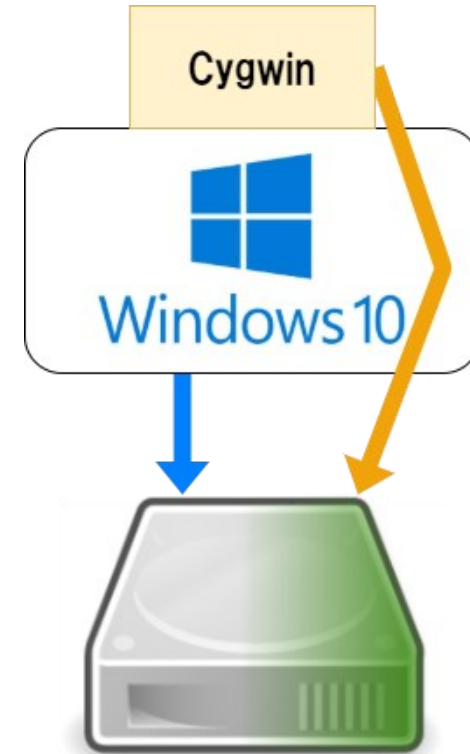
## ② WSL2とCygwinについて

### 一般的なLinux疑似環境



異なるディスク領域を使用

### CygwinによるLinux疑似環境



同じディスク領域を使用

- Cygwinは一般的な疑似環境と事なり、Windowsのディスク領域上に直接インストールされるため、Windows上のデータファイルにCygwin経由でもアクセスすることが可能になります(便利です)。

## ② WSL2とCygwinについて

### Linux疑似環境(Cygwinを含む)

操作(コマンド入力)



Cygwin

Windowsの命令に  
変換



Windowsの命令



PC(ハードウェア)



Windowsとしてプログラムを実行

### WSL2による仮想Linux環境

操作(コマンド入力)



ディストリビューション  
(Ubuntu など)



Linuxの命令



PC(ハードウェア)

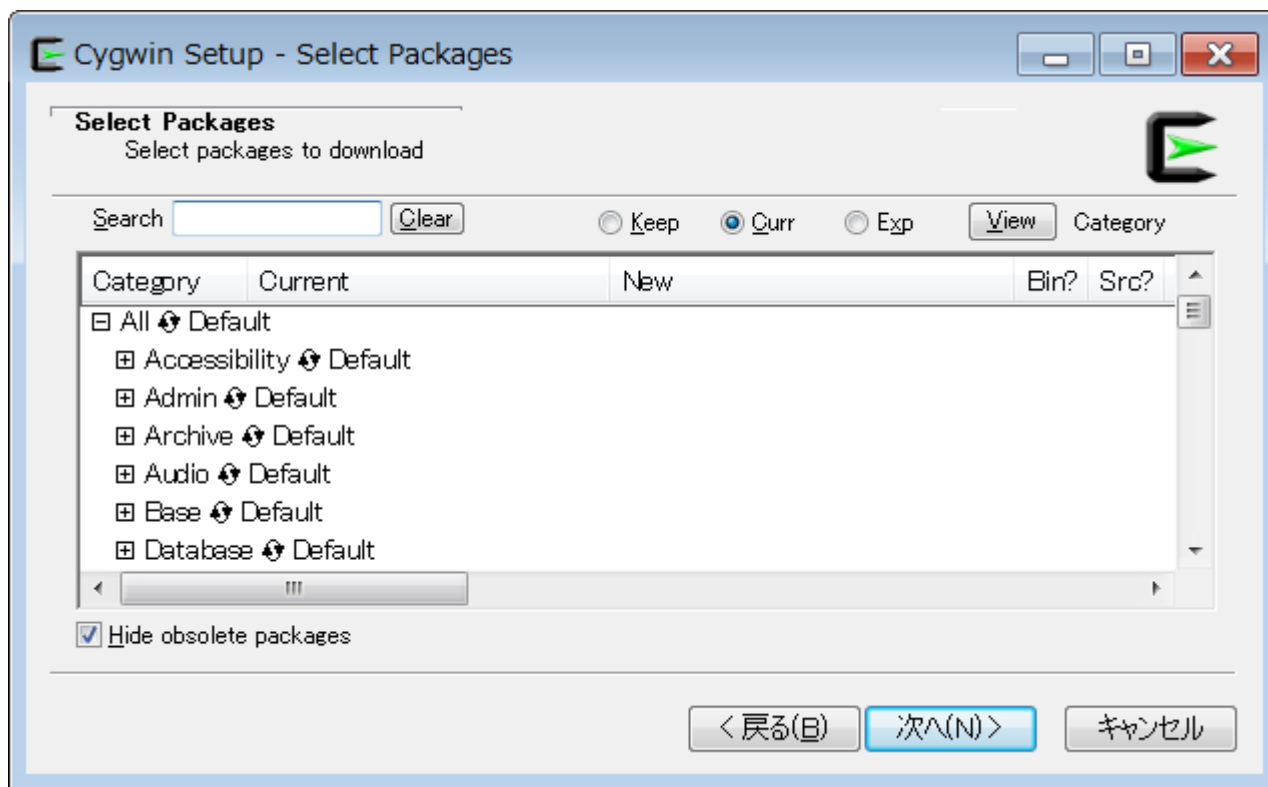


Linux(Kernel)としてプログラムを実行

- WSL2は疑似Linuxソフトウェアと違い、LinuxのOSそのものを仮想環境として構築し、一つのPCにWindowsとLinuxが同時に起動させます。
- WSL2は原理上、通常のLinuxと100%の互換性があります。

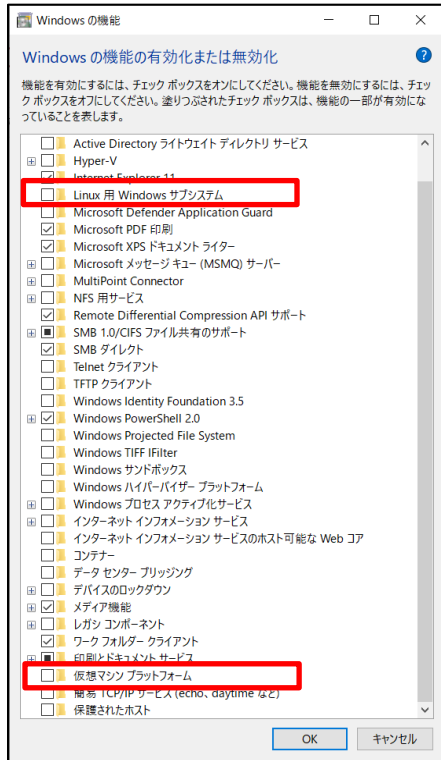


## ② WSL2とCygwinについて

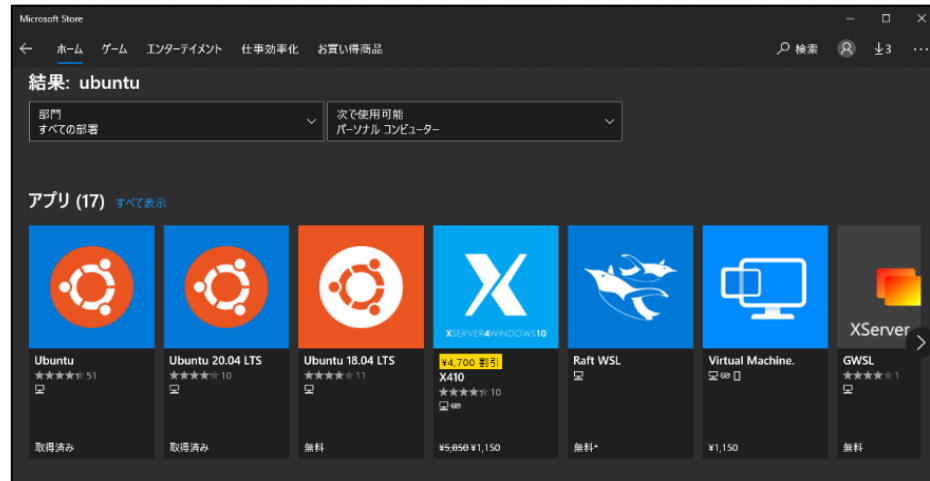


- Cygwinはインストールの過程で色々なパッケージの設定ができます。
- "Default"のままでも基本的機能は実装されますが、全機能を実装したい場合は、"Install"を選択します(完了まで長時間かかります)。
- Perl、Python、Rubyといったプログラミング言語も、Cygwin本体と同時にインストールすることができます。

## ② WSL2とCygwinについて



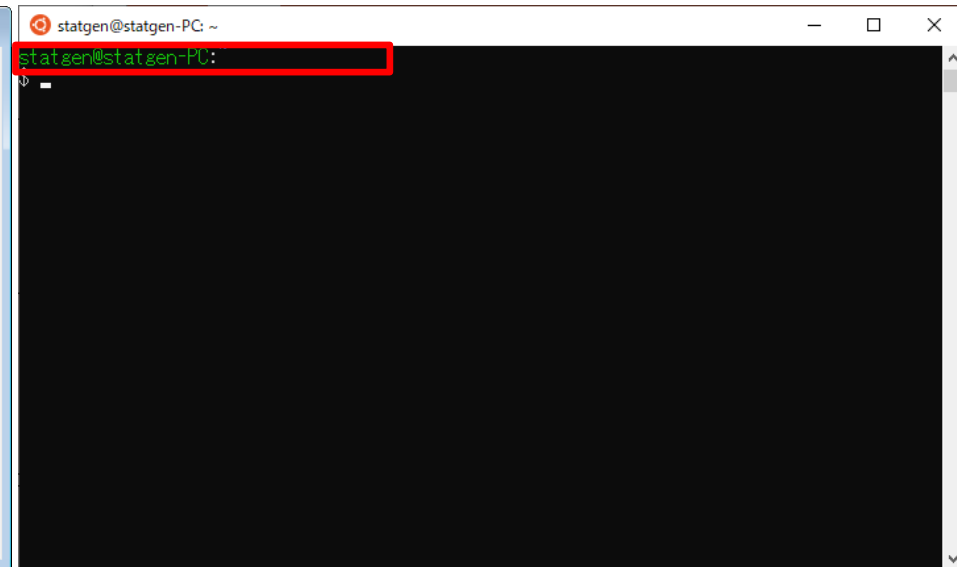
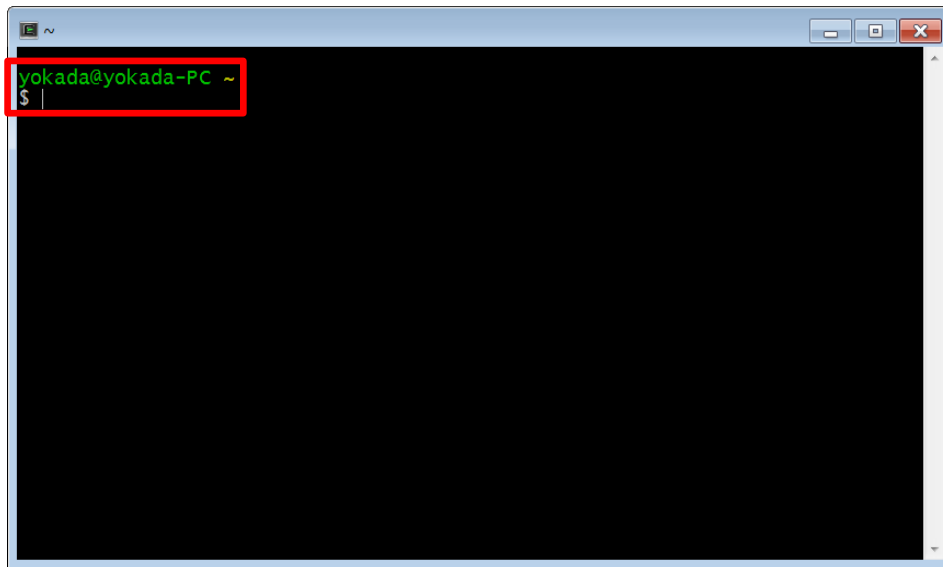
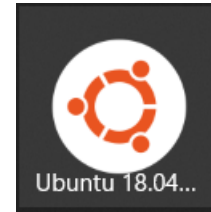
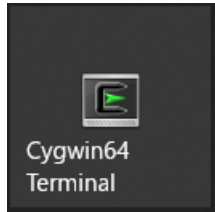
Windows側の設定



WSL2の有効化/ディストリビューションのインストール

- WSL2は仮想環境・Linuxディストリビューション・パッケージ・プログラミング言語等のインストールや環境構築を自分で実施する必要があります。
- ゲノムデータ解析では主要ソフトウェアがLinux版で配布されており、夏の学校では2021年から、より実戦的なWSL2を使用することにしました。(WSL2が導入できない場合、2020年の夏の学校を参考にCygwinを使って下さい<sup>18</sup>)

## ② WSL2とCygwinについて



- インストール後、各アイコンをクリックするとCygwin、WSL2が起動します。
- ここではWSL2に”Ubuntu 18.04-LTS”というディストリビューションをインストールしたので、アイコンはUbuntuとなっています。
- この画面(=Shell)でLinuxコマンドを打ち込むと実行されます。
- WSL2はソフトウェア(例:Python)を追加していく必要があります。

## Linux

- ① Linuxについて
- ② WSL2とCygwinについて
- ③ Linuxコマンド実習

## ③ Linuxコマンド実習



Linuxは、どのように使えばいいのですか？



Linuxコマンドをキーボードから打ち込んで使います。

- LinuxはCUIインターフェースのため、コンピューターへの命令をキーボード入力で行う必要があります。
- **Linuxコマンド**という命令文を打ち込むことで、実行します。

### ③ Linuxコマンド実習

メモリ使用状況

システム負荷状況

```
statgen@statgen-PC:~$ top - 15:21:03 up 22 min, 0 users, load average: 0.07, 0.04, 0.00
Tasks: 7 total, 1 running, 6 sleeping, 0 stopped, 0 zombie
%Cpu(s): 0.0 us, 0.0 sy, 0.0 ni, 100.0 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
KiB Mem : 52357732 total, 51887364 free, 399468 used, 70900 buff/cache
KiB Swap: 13631488 total, 13631488 free, 0 used, 51546404 avail Mem

  PID USER      PR  NI   VIRT   RES   SHR  S  %CPU  %MEM     TIME+ COMMAND
    1 root        20   0    892    584   516  S   0.0   0.0   0:00.03 init
     8 root        20   0    892     88    20  S   0.0   0.0   0:00.00 init
     9 root        20   0    892     88    20  S   0.0   0.0   0:00.14 init
    10 ysuenari   20   0  23116   4844  3252  S   0.0   0.0   0:00.13 bash
    65 root        20   0  63476   3680  3216  S   0.0   0.0   0:00.01 su
    66 statgen    20   0  22880   4832  3376  S   0.0   0.0   0:00.02 bash
    77 statgen    20   0  42108   3676  3212  R   0.0   0.0   0:00.00 top
```

個別のCPU実施プロセスの状況

- Linuxコマンド“top”を入力してみましょう。
- “top” は、OSの実施状況をリアルタイムで表示するコマンドです。
- “Ctrl + C”を入力することで、top実行画面を中止することができます。

### ③ Linuxコマンド実習

コマンド	機能
top	システム使用状況を表示
pwd	カレントディレクトリを表示
cd	カレントディレクトリを変更
ls	ディレクトリやファイル情報を表示
echo	文字列や変数の値を標準出力に表示
cp	ファイルやディレクトリをコピーする
mv	ファイルやディレクトリを移動する
rm	ファイルやディレクトリを消去する
wc	テキストファイルの大きさや行数を表示
gzip	ファイルとgz形式で圧縮
gunzip	gz形式のファイルを解凍
less	ファイル内容を一画面ごとに表示
head	ファイル内容を行単位で先頭からを表示
tail	ファイル内容を行単位で末尾からを表示

• Linuxコマンドは多数存在します。頻回に使用するものから覚えましょう。

### ③ Linuxコマンド実習

コマンド	機能
cut	ファイル内容を列単位で表示
grep	ファイル内容に含まれる文字列を検索
sed	ファイル内容に含まれる文字列を変換
diff	複数ファイルの内容を比較
cat	ファイル内容を表示・ファイルを縦に(行単位で)結合
zcat	gz形式圧縮ファイル内容を表示
paste	ファイルを横に(列単位で)結合
history	入力コマンド履歴を表示
wget	インターネット経由でファイルをダウンロード
exit	Linuxの終了
>	標準出力への表示内容をファイルに書き込む
	コマンド同士を連結する
&	コマンドをバックグラウンドで実行する
*	ファイルやディレクトリ名の一部を省略する

•Linuxコマンドは多数存在します。頻回に使用するものから覚えましょう。<sup>24</sup>



# ③ Linuxコマンド実習



The image shows a Google search interface with the search term "Linux コマンド". Below the search bar, there are navigation tabs: "すべて", "ショッピング", "ニュース", "画像", "動画", "もっと見る", and "検索ツール". The search results show approximately 426,000 results in 0.51 seconds. The first result is "Linuxコマンド 一覧 (UNIX入門)" from www.k4.dion.ne.jp/~mms/unix/linux\_com/, which explains Linux commands and lists Ls, Cp, Vi, and Rm. The second result is "2時間でマスター！ Linux/Unixコマンドの 使い方超入門 | 侍エンジニア塾 ..." from www.sejuku.net/blog/2643, discussing the need for Linux knowledge in web programming. The third result is "Linux入門 - コマンドリファレンス - Webkaru" from webkaru.net, providing a command reference for Linux.

•Linuxコマンドを解説した書籍やWebサイトも多数存在します。

### ③ Linuxコマンド実習



Linuxコマンドを覚えるのが大変そうなのですが・・・



実際に手を動かして、コマンドを打ち込んでみると習得が速いです。

- 最初は覚えるのが難しく感じますが、**実際に手を動かして、コマンドを打ち込んでみる**と習得がはやいです(経験則ですが・・・)。
- 教本を眺めているだけだと、なかなか習得が進まないです。

### ③ Linuxコマンド実習

```
statgen@statgen-PC: ~
```

```
$ pwd  
/home/statgen
```

※Shellを起動して、\$マークの後に続くコマンドを、実際に入力してみましょう。

入力するコマンド

```
statgen@statgen-PC: ~
```

```
$ cd /mnt/c/SummerSchool/
```

※Cygwinの場合 /mnt/を/cygdrive/に変えてください。

※Macユーザーの方は、演習ファイルを置いたディレクトリを適宜指定して実行してください。

```
statgen@statgen-PC: /mnt/c/SummerSchool
```

```
$ pwd
```

```
/mnt/c/SummerSchool
```

```
statgen@statgen-PC: /mnt/c/SummerSchool
```

```
$ ls
```

```
Genetics GeneomeDataAnalysis3 GeneomeDataAnalysis6 ...
```

• **”pwd”**でカレントディレクトリの表示、**”cd”**でディレクトリの移動、**”ls”**でディレクトリ内のファイルの表示、ができます。

### ③ Linuxコマンド実習

```
statgen@statgen-PC: /mnt/c/SummerSchool
```

```
$ cd ..
```

```
$ pwd
```

```
/mnt/c
```

```
statgen@statgen-PC: /mnt/c/
```

```
$ cd
```

```
$ pwd
```

```
/home/statgen
```

- **"cd .."** (もしくは**"cd ../"**)で一つ上のディレクトリに、**"cd"**のみでホームディレクトリに、移動することができます。

### ③ Linuxコマンド実習

```
statgen@statgen-PC: /mnt/c/SummerSchool
```

```
$ ls -la
```

```
合計 16
```

```
d-----+ 1 statgen None 0 8月 14 21:45 .  
d-----+ 1 statgen None 0 8月 14 21:43 ..  
d-----+ 1 statgen None 0 8月 15 16:17 Linux  
d-----+ 1 statgen None 0 8月 15 00:12 Rsoftware
```

- “ls -la”と、オプションを指定すると、より詳細な情報が得られます。
- コマンドだけでなく、主要なオプションを覚えることも重要です。
- Linuxでは、半角スペースは、コマンド、オプション、ファイル名等の区切り文字として認識されます。名前に半角スペースを含むファイルは扱いが難しいため、避けることが推奨されます。

# ③ Linuxコマンド実習

```
statgen@statgen-PC: /mnt/c/SummerSchool
```

```
$ ls -la
```

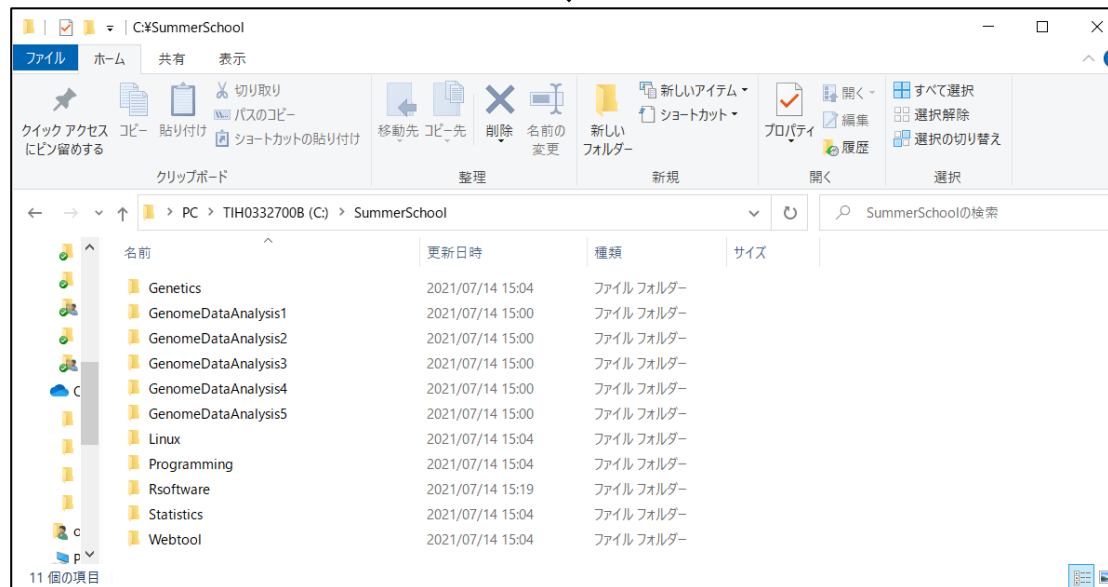
```
合計 16
```

```
d-----+ 1 statgen None 0 8月 14 21:45 .
```

```
d-----+ 1 statgen None 0 8月 14 21:43 ..
```

```
d-----+ 1 statgen None 0 8月 15 16:17 Linux
```

```
d-----+ 1 statgen None 0 8月 15 00:12 Rsoftware
```



• Windows上のファイルを、WSL2を通して閲覧していることになります。

## ③ Linuxコマンド実習

LS(1) User Commands LS(1)

NAME

ls - list directory contents

SYNOPSIS

ls [OPTION]... [FILE]...

DESCRIPTION

List information about the FILEs (the current directory by default). Sort entries alphabetically if none of -cftuvSUX nor --sort is specified.

Mandatory arguments to long options are mandatory for short options too.

-a, --all

• ”man ls”のように、”man”をコマンド名と共に入力することで、オプション詳細がわかります。

(初めは、入門本やWebページを参考にする方がわかりやすいです。)

### ③ Linuxコマンド実習

※Cygwinの場合 /mnt/を/cygdrive/に変えてください

ここまで入力した段階で、タブキーを押すと・・・

```
statgen@statgen-PC: ~
```

```
$ cd /mnt/c/SummerSchool/L
```



残りが自動的に補完される！

```
statgen@statgen-PC: ~
```

```
$ cd /mnt/c/SummerSchool/Linux/
```



- ディレクトリ名やファイル名を入力する際、初めの数文字を入力した段階でタブキーを押すと、残りの文字を自動的に補完してくれます。



### ③ Linuxコマンド実習

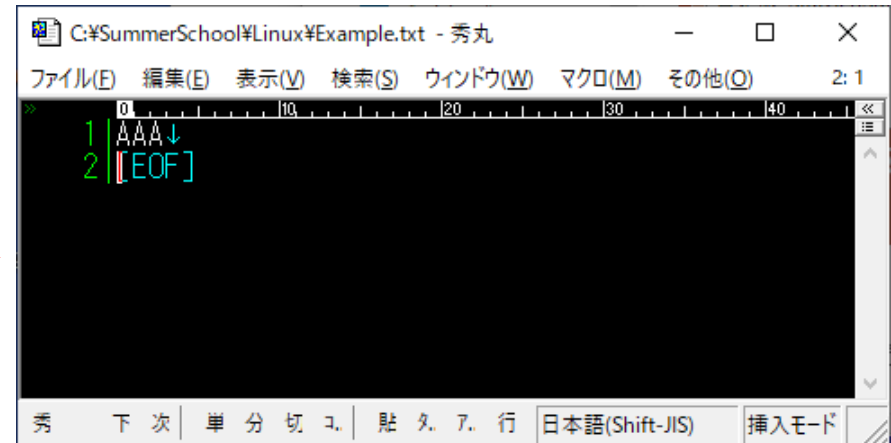
```
statgen@statgen-PC: /mnt/c/SummerSchool/Linux
```

```
$ echo "AAA"
```

```
AAA
```

```
statgen@statgen-PC: /mnt/c/SummerSchool/Linux
```

```
$ echo "AAA" > Example.txt
```



```
C:\SummerSchool\Linux\Example.txt - 秀丸
ファイル(E) 編集(E) 表示(V) 検索(S) ウィンドウ(W) マクロ(M) その他(O) 2: 1
1 | AAA↓
2 | [EOF]
```

- **“echo”**で、後に続く文字列や変数の値を、標準出力(Linuxコマンド画面)に表示することができます。
- 標準出力に表示される内容は、**“>”**を使うと、後に指定されるファイルに書き出すことができます。

### ③ Linuxコマンド実習

statgen@statgen-PC: ~

※Cygwinの場合 /mnt/を/cygdrive/に変えてください

\$ echo "AAA" > Example.txt

十字キーの↑を押すと…

statgen@statgen-PC: ~

\$

一つ前に入力したコマンドを呼び出せる！

↑を押した数だけ、さかのぼって呼び出しが可能！

statgen@statgen-PC: ~

\$ echo "AAA" > Example.txt

少し修正したコマンドを容易に実行！

statgen@statgen-PC: ~

\$ echo "AAA" > Example2.txt

- 十字キーの↑を押すと、一つ前に入力したコマンドが表示されるため、同じコマンドを入力する手間が省けます。
- 一つ前に入力したコマンドを表示した上で、少し修正を加えていくことで、入力コマンドを少しずつアップデートしていくことができます。

### ③ Linuxコマンド実習

```
statgen@statgen-PC: /mnt/c/SummerSchool/Linux
```

```
$ echo "AAA BBB"
```

```
AAA BBB
```

```
statgen@statgen-PC: /mnt/c/SummerSchool/Linux
```

```
$ echo "AAA\tBBB\nCCC"
```

```
AAA\tBBB\nCCC
```

```
statgen@statgen-PC: /mnt/c/SummerSchool/Linux
```

```
$ echo -e "AAA\tBBB\nCCC"
```

```
AAA   BBB
```

```
CCC
```

Mac OSでは、「¥」は「バックスラッシュ」になります。

- "echo"では、**タブ(¥t)**や**改行(¥n)**といった特殊文字も表示できます。
- その際、**オプション"-e"**を付ける必要があります。

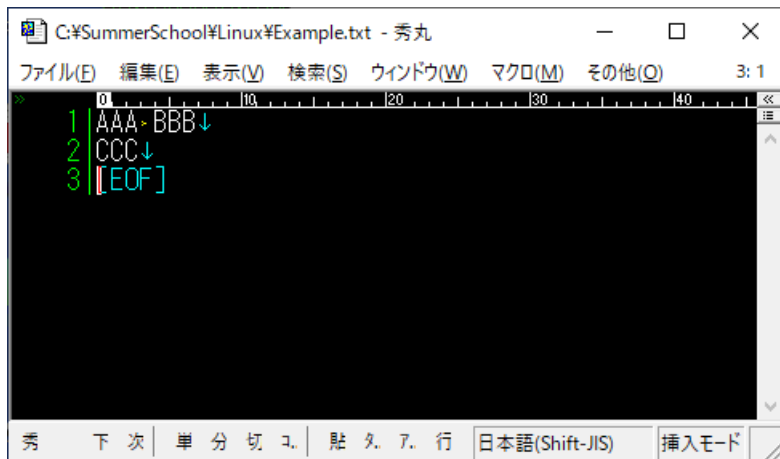
### ③ Linuxコマンド実習

```
statgen@statgen-PC: /mnt/c/SummerSchool/Linux
```

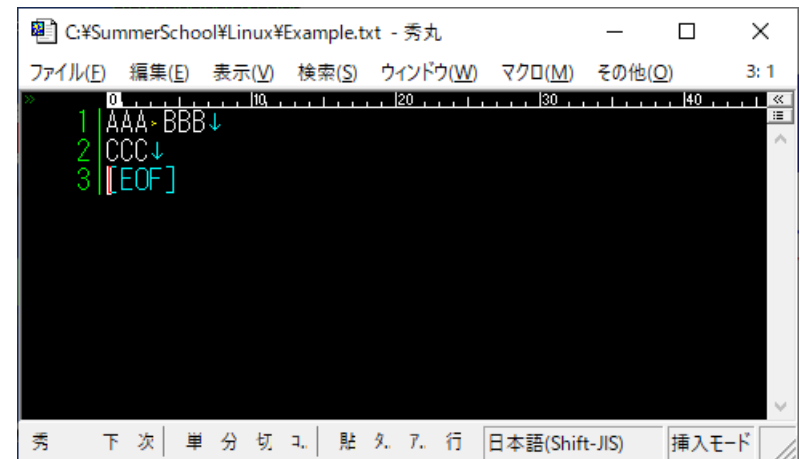
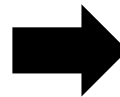
```
$ echo -e "AAA\tBBB\nCCC" > Example.txt
```

```
statgen@statgen-PC: /mnt/c/SummerSchool/Linux
```

```
$ cp Example.txt Example2.txt
```



```
C:\SummerSchool\Linux\Example.txt - 秀丸
ファイル(E) 編集(E) 表示(V) 検索(S) ウィンドウ(W) マクロ(M) その他(O) 3: 1
1 AAA·BBB↓
2 CCC↓
3 [EOF]
```



```
C:\SummerSchool\Linux\Example.txt - 秀丸
ファイル(E) 編集(E) 表示(V) 検索(S) ウィンドウ(W) マクロ(M) その他(O) 3: 1
1 AAA·BBB↓
2 CCC↓
3 [EOF]
```

• ”cp”でファイルやディレクトリをコピーすることができます。

### ③ Linuxコマンド実習

```
statgen@statgen-PC: /mnt/c/SummerSchool/Linux
```

```
$ mv Example.txt /mnt/c/SummerSchool/
```

※Cygwinの場合 /mnt/を/cygdrive/に変えてください

```
statgen@statgen-PC: /mnt/c/SummerSchool/Linux
```

```
$ cd .. ← 1つ上の階層のディレクトリを意味する
```

```
statgen@statgen-PC: /mnt/c/SummerSchool
```

```
$ ls
```

```
Example.txt Linux Rsoftware
```

```
statgen@statgen-PC: /mnt/c/SummerSchool
```

```
$ rm Example.txt
```

```
statgen@statgen-PC: /mnt/c/SummerSchool
```

```
$ ls
```

```
Linux Rsoftware
```

- **”mv”**でファイルやディレクトリを移動することができます。
- **”rm”**でファイルやディレクトリを消去することができます。

### ③ Linuxコマンド実習



rmコマンドで消去したファイルは復活できますか？



Linux上で消去したファイルは、復元できません。



- Linux上で消去したファイルは復元できません。
- rmコマンドを使用する場合、入力前に最終確認してから行いましょう。
- 解析の経過で生成した中間ファイルは、(ディスク容量に余裕がある場合は) 消去せず保存しておくとおと安心です。

### 入力してはいけないコマンド

“rm \*”

#### よくある失敗例

“\*”は任意の全ての文字列を指定し、全ファイル・全ディレクトリが対象に

“rm \* .txt”

誤って半角スペースが挿入されていることに、  
(リターンキーを押してから)気づく...

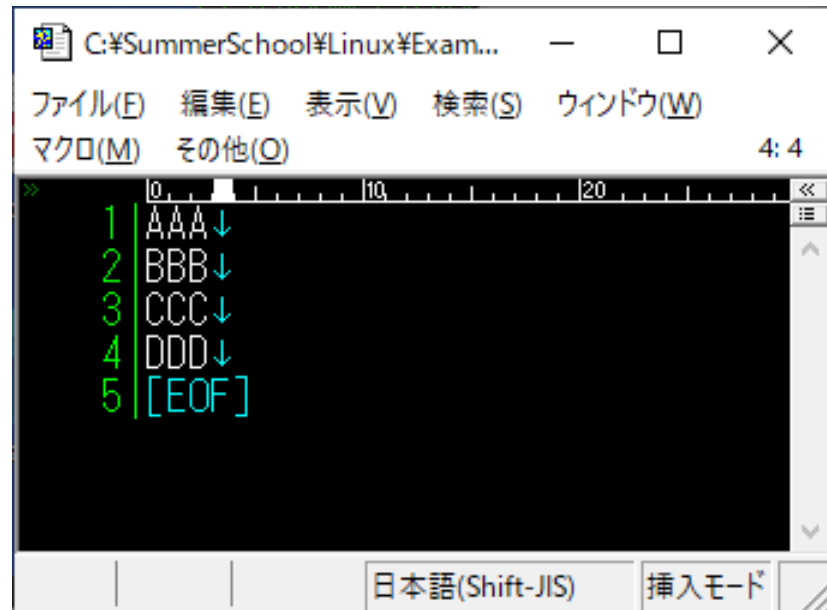
- 同じコマンドライン上に、“rm”と“\*”が並んでいる時は、細心の注意を払いましょう。
- エンターキーを押す前に、一呼吸いれる習慣がオススメです。

### ③ Linuxコマンド実習

```
statgen@statgen-PC: /mnt/c/SummerSchool/Linux  
$ echo -e "AAA¥nBBB¥nCCC¥nDDD" > Example.txt
```

```
statgen@statgen-PC: /mnt/c/SummerSchool/Linux  
$ wc Example.txt  
4 4 16 Example.txt
```

行数が表示される



```
C:\¥SummerSchool¥Linux¥Exam...  
ファイル(E) 編集(E) 表示(V) 検索(S) ウィンドウ(W)  
マクロ(M) その他(O) 4: 4  
1 AAA↓  
2 BBB↓  
3 CCC↓  
4 DDD↓  
5 [EOF]
```

• ”wc”で、テキストファイルの大きさ(行数)を確認することができます。<sup>40</sup>



### ③ Linuxコマンド実習

```
statgen@statgen-PC: /mnt/c/SummerSchool/Linux  
$ echo -e "AAA¥nBBB¥nCCC¥nDDD" > Example.txt
```

```
statgen@statgen-PC: /mnt/c/SummerSchool/Linux  
$ gzip Example.txt
```

```
statgen@statgen-PC: /mnt/c/SummerSchool/Linux  
$ ls  
Example.txt.gz
```

```
statgen@statgen-PC: /mnt/c/SummerSchool/Linux  
$ gunzip Example.txt.gz
```

```
statgen@statgen-PC: /mnt/c/SummerSchool/Linux  
$ ls  
Example.txt
```

- **“gzip”**でファイルのgz形式の圧縮、**“gunzip”**でgz形式ファイルの解凍ができます。

### ③ Linuxコマンド実習

```
statgen@statgen-PC: /mnt/c/SummerSchool/Linux  
$ echo -e "AAA¥nBBB¥nCCC¥nDDD" > Example.txt
```

```
statgen@statgen-PC: /mnt/c/SummerSchool/Linux  
$ less Example.txt
```

AAA

BBB

CCC

DDD

- **“less”**でファイル内容を一画面ごとに表示することができます。
- **“less”**の表示画面から戻る際は、**“q”**を押します。
- **大きい(長い)ファイルの内容を確認するとき**に便利です。

### ③ Linuxコマンド実習

```
statgen@statgen-PC: /mnt/c/SummerSchool/Linux
$ echo -e "AAA¥nBBB¥nCCC¥nDDD" > Example.txt
```

```
statgen@statgen-PC: /mnt/c/SummerSchool/Linux
$ head -n 2 Example.txt
AAA
BBB
```

```
statgen@statgen-PC: /mnt/c/SummerSchool/Linux
$ tail -n 2 Example.txt
CCC
DDD
```

- **“head” / “tail”**で、ファイル内容を行単位で先頭 / 末尾から表示することができます。
- **“-n”オプション**で、表示する行数を指定する必要があります。

### ③ Linuxコマンド実習

```
statgen@statgen-PC: /mnt/c/SummerSchool/Linux
```

```
$ echo -e "AA1\tAA2\tAA3\nBB1\tBB2\tBB3" > Example.txt
```

```
statgen@statgen-PC: /mnt/c/SummerSchool/Linux
```

```
$ cut -f 2 Example.txt
```

```
AA2
```

```
BB2
```

```
statgen@statgen-PC: /mnt/c/SummerSchool/Linux
```

```
$ cut -f 1-3 Example.txt
```

```
AA1  AA2  AA3
```

```
BB1  BB2  BB3
```

- **”cut”**で、ファイル内容を列単位で表示することができます。
- **”-f”オプション**で、表示する列の場所を指定する必要があります。

### ③ Linuxコマンド実習

```
statgen@statgen-PC: /mnt/c/SummerSchool/Linux
```

```
$ echo -e "AA1\tAA2\tAA3\nBB1\tBB2\tBB3" > Example.txt
```

```
statgen@statgen-PC: /mnt/c/SummerSchool/Linux
```

```
$ grep "A" Example.txt
```

```
AA1   AA2   AA3
```

```
statgen@statgen-PC: /mnt/c/SummerSchool/Linux
```

```
$ grep "1" Example.txt
```

```
AA1   AA2   AA3
```

```
BB1   BB2   BB3
```

- ”grep”で、ファイル内容に含まれる文字列を検索することができます。

### ③ Linuxコマンド実習

```
statgen@statgen-PC: /mnt/c/SummerSchool/Linux
```

```
$ echo -e "AA1\tAA2\nBB1\tBB2" > Example.txt
```

```
statgen@statgen-PC: /mnt/c/SummerSchool/Linux
```

```
$ cat Example.txt
```

```
AA1  AA2
```

```
BB1  BB2
```

```
statgen@statgen-PC: /mnt/c/SummerSchool/Linux
```

```
$ cat Example.txt Example.txt
```

```
AA1  AA2
```

```
BB1  BB2
```

```
AA1  AA2
```

```
BB1  BB2
```

- **"cat"**で、ファイルの内容を表示したり、ファイルを縦に(行単位で)結合することができます。

### ③ Linuxコマンド実習

```
statgen@statgen-PC: /mnt/c/SummerSchool/Linux
```

```
$ echo -e "AA1\tAA2\nBB1\tBB2" > Example.txt
```

```
statgen@statgen-PC: /mnt/c/SummerSchool/Linux
```

```
$ gzip Example.txt
```

```
statgen@statgen-PC: /mnt/c/SummerSchool/Linux
```

```
$ cat Example.txt.gz
```

```
WExample.txtst4tt4rr2tr2du ← 読めない・・・
```

```
statgen@statgen-PC: /mnt/c/SummerSchool/Linux
```

```
$ zcat Example.txt.gz
```

```
AA1  AA2  
BB1  BB2 ← ちゃんと読める！！
```

※Mac OSの場合、zcatではなくgzcatコマンドになります。

• **“zcat”**で、gz形式に圧縮されたファイルの内容を表示することができます。gunzipで解凍する手間を省くことができます。

### ③ Linuxコマンド実習

```
statgen@statgen-PC: /mnt/c/SummerSchool/Linux
```

```
$ echo -e "AA1¥tAA2¥nBB1¥tBB2" > Example.txt
```

```
statgen@statgen-PC: /mnt/c/SummerSchool/Linux
```

```
$ paste Example.txt Example.txt
```

```
AA1  AA2  AA1  AA2
```

```
BB1  BB2  BB1  BB2
```

```
statgen@statgen-PC: /mnt/c/SummerSchool/Linux
```

```
$ paste -d "-" Example.txt Example.txt
```

```
AA1  AA2-AA1 AA2
```

```
BB1  BB2-BB1 BB2
```

- **“paste”**で、ファイルを横に(列単位で)結合することができます。
- 結合時の区切り文字はデフォルトでタブ(¥t)ですが、**“-d”オプション**で指定することもできます。



### ③ Linuxコマンド実習

```
statgen@statgen-PC: /mnt/c/SummerSchool/Linux
```

```
$ echo -e "AA1¥tAA2¥nBB1¥tBB2" > Example.txt
```

```
statgen@statgen-PC: /mnt/c/SummerSchool/Linux
```

```
$ paste Example.txt Example.txt | cut -f 2,4
```

```
AA2  AA2
```

```
BB2  BB2
```

```
statgen@statgen-PC: /mnt/c/SummerSchool/Linux
```

```
$ cat Example.txt Example.txt | tail -n 3 | cut -f 2
```

```
BB2
```

```
AA2
```

```
BB2
```

- **”|”(パイプ)**で、Linuxコマンドにより得られた標準出力への出力内容を、次のLinuxコマンドの入力内容として使うことができます。
- **複数の”|”**を使い、複数のLinuxコマンドをつなぎあわせることで、複雑な作業を一回のコマンド入力で実施可能になります。

### ③ Linuxコマンド実習

statgen@statgen-PC: /mnt/c/SummerSchool/Linux

\$ wget http://www.sg.med.osaka-u.ac.jp/images/header.png

```
--2016-08-15 17:59:04-- http://www.sg.med.osaka-u.ac.jp/images/header.png
www.sg.med.osaka-u.ac.jp (www.sg.med.osaka-u.ac.jp) をDNSに問いあわせています... 133.1.138.171
www.sg.med.osaka-u.ac.jp (www.sg.med.osaka-u.ac.jp)|133.1.138.171|:80 に接続しています... 接続しました。
HTTP による接続要求を送信しました、応答を待っています... 200 OK
長さ: 307671 (300K) [image/png]
`header.png' に保存中
header.png      100%[=====>] 300.46K  --.-KB/s 時間 0.004s
2016-08-15 17:59:04 (67.9 MB/s) - `header.png' へ保存完了 [307671/307671]
```

statgen@statgen-PC: /mnt/c/SummerSchool/Linux

\$ wget -q http://www.sg.med.osaka-u.ac.jp/images/header.png



※Mac OSの場合、wgetではなくcurlコマンドになります。

- **“wget”**で、インターネット上のファイルをダウンロードできます。
- **“-q”オプション**を使うと、ダウンロード経過を表示しなくなります。

### ③ Linuxコマンド実習

```
statgen@statgen-PC: /mnt/c/SummerSchool/Linux
```

```
$ wget -q http://www.sg.med.osaka-u.ac.jp/files/RA_GWASmeta_TransEthnic_v2.txt.gz
```

(Ctrl+C)で強制終了

```
statgen@statgen-PC: /mnt/c/SummerSchool/Linux
```

```
$ wget -q http://www.sg.med.osaka-u.ac.jp/files/RA_GWASmeta_TransEthnic_v2.txt.gz &  
[1] 9180
```

```
statgen@statgen-PC: /mnt/c/SummerSchool/Linux
```

```
$
```

```
[1]+ 終了
```

```
wget -q http://www.sg.med.osaka-u.ac.jp/files/RA_GWASmeta_TransEthnic_v2.txt.gz
```

- 実行中のコマンドを強制終了する場合、“Ctrl+C”を押します。
- コマンドの末尾に“&”を付けると、バックグラウンドで実行されるため、コマンド終了を待たずに別のコマンドを入力できるようになります。
- バックグラウンドで実行中のコマンドを確認する場合、“top”を使います。

### ③ Linuxコマンド実習

```
statgen@statgen-PC: /mnt/c/SummerSchool/Linux
```

```
$ exit
```

```
statgen@statgen-PC: /mnt/c/SummerSchool/Linux
```

```
$ history 10
```

```
162 gzip Example.txt
163 cat Example.txt.gz
164 zcat Example.txt.gz
165 echo -e "AA1¥tAA2¥nBB1¥tBB2" > Example.txt
166 paste Example.txt Example.txt
167 paste -d "-" Example.txt Example.txt
168 echo -e "AA1¥tAA2¥nBB1¥tBB2" > Example.txt
169 paste Example.txt Example.txt | cut -f 2,4
170 cat Example.txt Example.txt | tail -n 3 | cut -f 2
171 wget http://www.sg.med.osaka-u.ac.jp/images/header.png
```

```
statgen@statgen-PC: /mnt/c/SummerSchool/Linux
```

```
$ history 10 > Log.txt
```

- **“exit”**でLinux端末(本講義ではUbuntuやCygwinのShell画面)を終了できます。
- コマンド入力履歴を**“history”**で確認できます。**“history”**の出力をログファイルに保存すると、コマンド入力履歴を手軽に保存できます。

# 終わりに

- Linuxの使い方を、駆け足でなぞってみました。
- Linuxが難しくないこと、便利であることを感じてもらえればと思います。
- ゲノムデータ解析の分野では、主な解析はLinux上で実施されています。
- 今回の講義は、一般ユーザーが使うコマンドに絞って解説しています。
- サーバー管理コマンドや、ユーザーアカウント管理コマンド、パスの概念、ネットワーク関連の機能、については解説していません。必要に応じて、各自勉強して頂ければと思います。