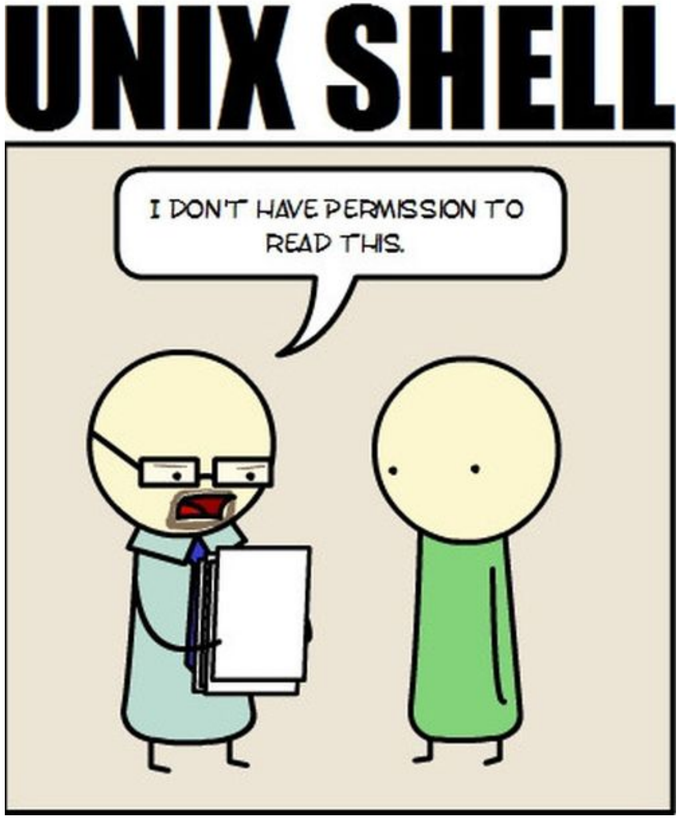
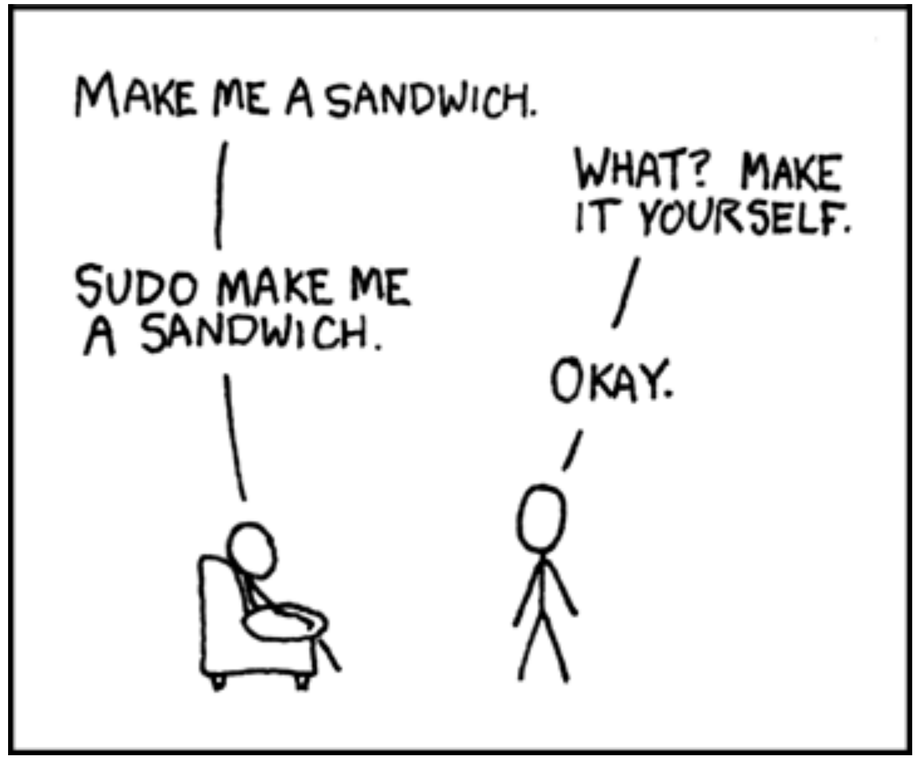
# Data analysis with the shell

Spring 2021
PCfB Class 3
January 29, 2021



# Outline

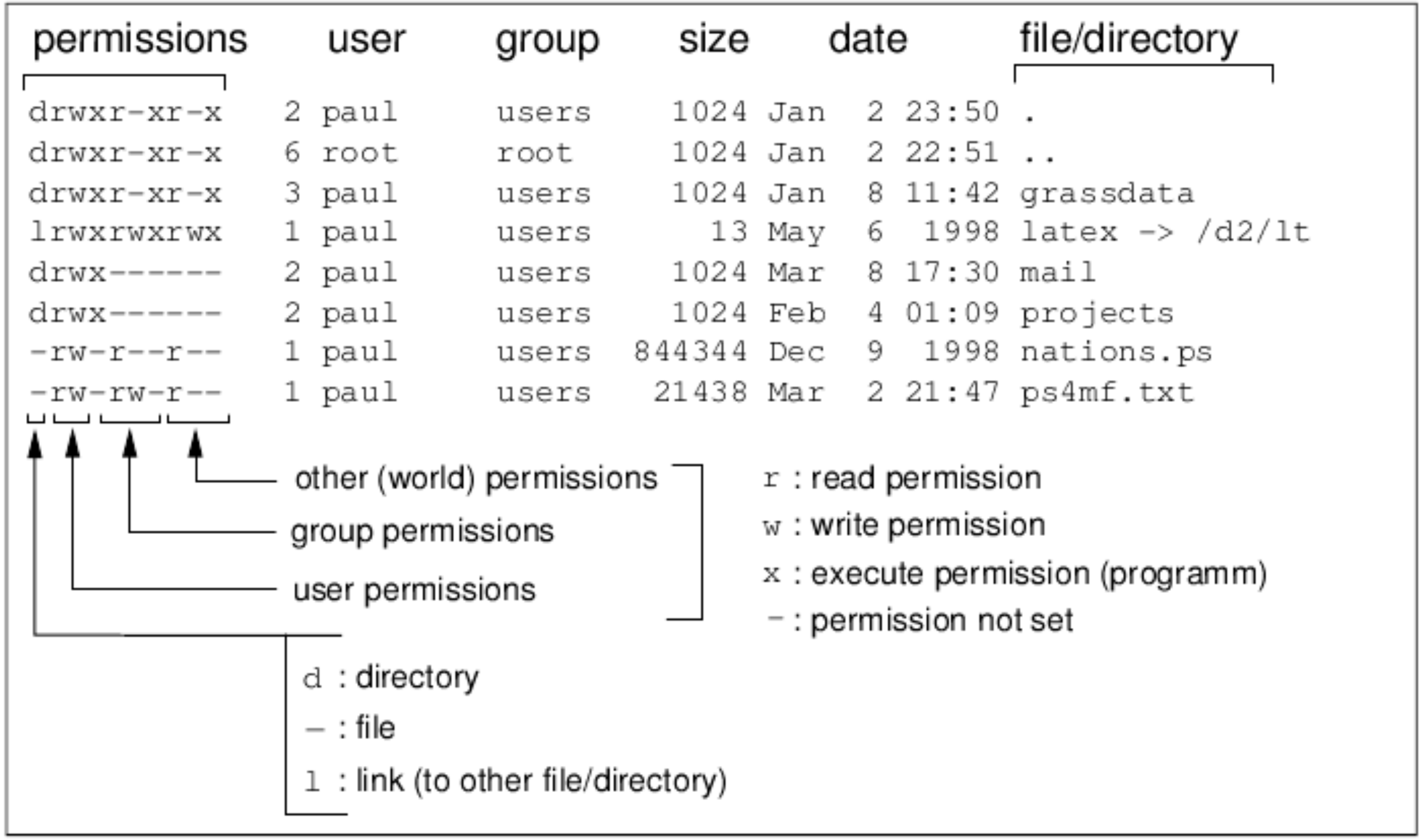
- Installing/running programs
- ‘for’ loops
- Shell scripts
- Exercises

# Installing/running programs

# Computers only understand their native machine language

# 2 types of programming languages

# Permissions (‘ls -l’)



# Changing Permissions

## `chmod` (change mode)

Add execute for User: `chmod u+x file.txt`

Add read and write for Group: `chmod g+rw file.txt`

Remove write and execute for Other: `chmod o-wx file.txt`

All three in one command: `chmod u+x,g+rw,o-wx file.txt`

# $PATH

- A list of directories
  - Locations your computer looks for command-line software
- Searched in the order listed
- To view: `echo $PATH`
- To add a directory: `PATH="$PATH:path/to/new/dir"`

# Recommendation

- Create three directories in your home directory:
  - *scripts* (your own custom scripts, PCfB p. 85-88)
  - *programs* (code downloaded from others)
  - *source* (pre-compiled source code)
- Add *scripts* and *programs* to $PATH

# Dependencies

# Installing program demo

# ‘for’ loops

# `for` loop

- Simple, but powerful way to repeatedly execute the same commands for different files, parameter values, etc.
- Can be included in scripts or run directly on command line

# Basic syntax

```
for file in *.sh; do chmod u+x $file; done
```

# ‘for’ loop examples

```
for file in *.fasta;
do print_fasta_seq_lengths.py $file;
done
```

```
for file in *txt; do cp $file copy_$file;
mkdir dir_$file; mv $file dir_$file; done
```

# ‘for’ loop demo

# Shell scripts

# Why use shell scripts?

- Automate a series of commands
  - particularly useful when each command takes a long time to run
- Creates a record of commands that have been run
- Easy format for rerunning commands

# Two ways to specify the interpreter to use

# Specify interpreter inside script

```
1  #!/bin/bash¬
2  ¬
3  mkdir test¬
4  cd test¬
```

# File extensions

- Recommended (but not required) to save script with specific file extension
- Allows recognition from file name
- Syntax-specific coloring in text editors
- For shell script:

# Shell + Regexp method

1. Use the shell to generate a list of files/directories
2. Use regular expressions within your text editor to turn those file/directory names into a list of commands

# Shell script demo

# Assignment