

Practical Bioinformatics

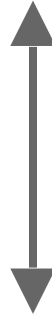
Basic Unix

/ˈyōoniks/

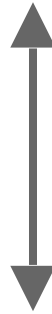
A widely used multiuser operating system.



HARDWARE
(CPU, MEMORY, DEVICES)

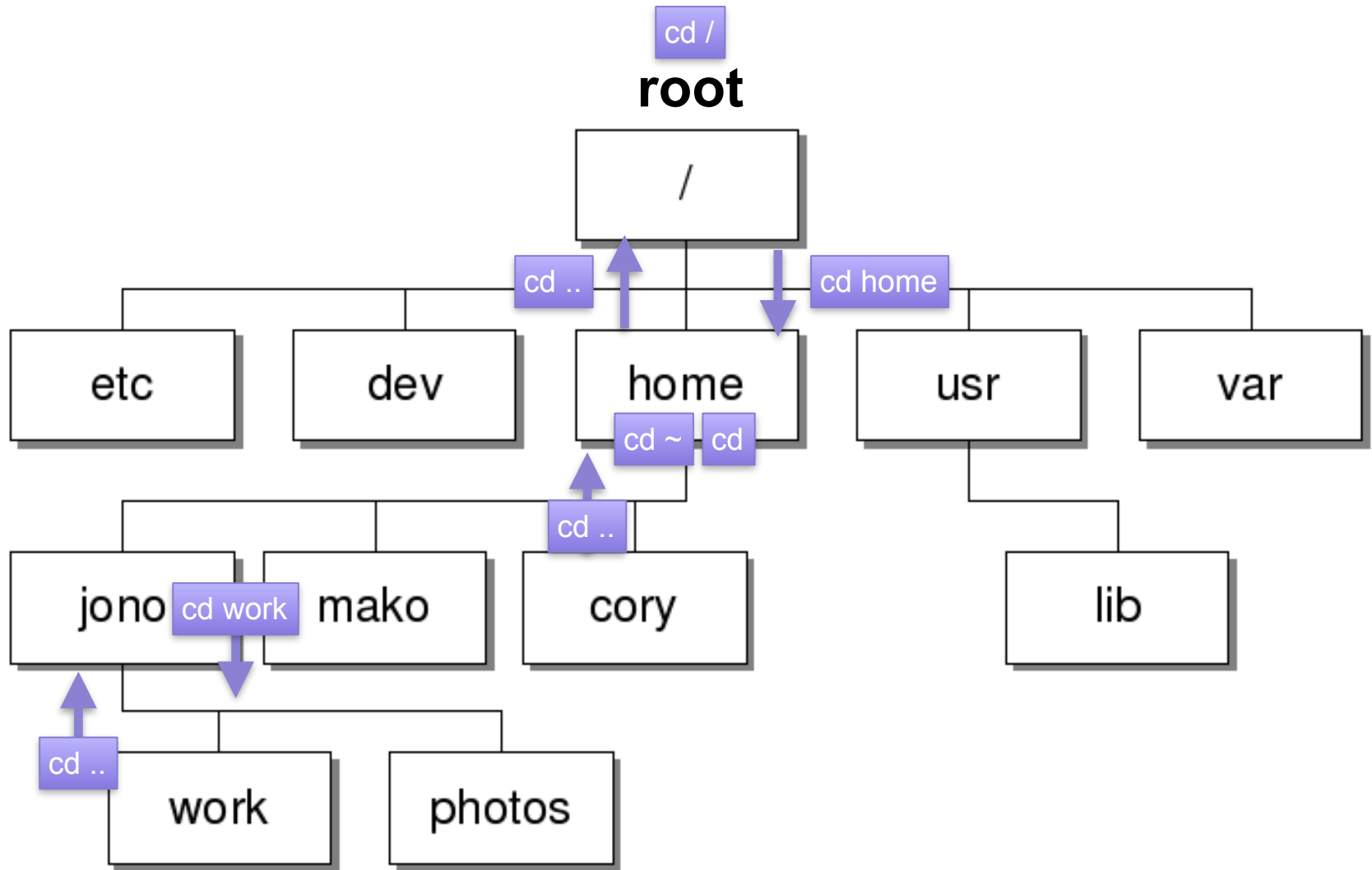


KERNEL
(operating system)



PROGRAMS

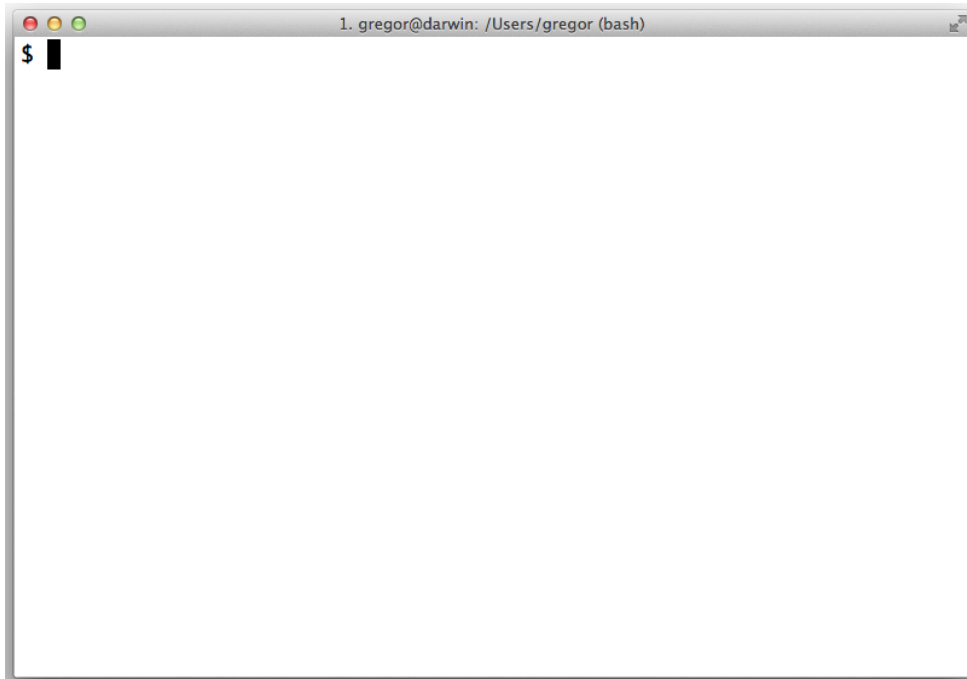
Filesystem structure



Shell

The most generic sense of the term **shell** means any program that users employ to type commands.

A **shell** is software that provides an interface for users of an operating system to access the services of a kernel.



CLI: command line (once the only way to control the computer)

GUI: graphical user interface

Shell basics

```
$ whoami      # what is my username on this host
$ uname -a    # show basic info of the host OS
$ df -h
```

Filesystem	Size	Used	Avail	Use%	Mounted on
overlay	282G	102G	169G	38%	/
tmpfs	64M	0	64M	0%	/dev
tmpfs	16G	0	16G	0%	/sys/fs/cgroup
/dev/vda1	282G	102G	169G	38%	/etc/hosts
shm	64M	0	64M	0%	/dev/shm
tmpfs	16G	0	16G	0%	/proc/acpi
tmpfs	16G	0	16G	0%	/proc/scsi
tmpfs	16G	0	16G	0%	/sys/firmware

Home directory (~)

```
$ pwd          # where are we?
$ cd ~         # change directory to /home/user
$ pwd
$ cd ..        # go one level up (/Users)
$ pwd
$ cd           # same as cd ~
$ pwd         # we are home
```

~ = /home/username

Download exercise files

```
$ cd ~
$ curl http://expressrna.org/bioinfo.tar.gz -o bioinfo.tgz
$ tar xzf bioinfo.tgz
$ ls -l bioinfo
$ cd bioinfo
```

~/bioinfo = /home/rstudio/bioinfo

List directory content

```
$ ls -l
```

```
drwxr-xr-x 2 rstudio users 4096 May  2 17:15 dd
-rw-r--r-- 1 rstudio users   24 May  8 2013 file1.txt
-rw-r--r-- 1 rstudio users   23 May  8 2013 file2.txt
-rw-r--r-- 1 rstudio users 941506 Jan 28 2013 protein_abundance.txt
-rw-r--r-- 1 rstudio users   885 May 12 2016 proteins.fasta
-rw-r--r-- 1 rstudio users   328 May  8 2013 proteins.fasta.gz
-rwxr-xr-x 1 rstudio users    60 Apr 29 2015 script1.py
```

Use special characters (?, *)

```
$ ls -l file?.txt
```

```
-rw-r--r-- 1 rstudio users 24 May  8 2013 file1.txt
-rw-r--r-- 1 rstudio users 23 May  8 2013 file2.txt
```

```
$ ls -l protein*
```

```
-rw-r--r-- 1 rstudio users 941506 Jan 28 2013 protein_abundance.txt
-rw-r--r-- 1 rstudio users   885 May 12 2016 proteins.fasta
-rw-r--r-- 1 rstudio users   328 May  8 2013 proteins.fasta.gz
```


Copying files

```
$ cp proteins.fasta pcopy.fasta
```

```
$ mkdir temp
```

```
$ cp pcopy.fasta temp
```

Moving / renaming files

```
$ mv pcopy.fasta temp
```

```
$ mv temp temp2
```

Deleting files

```
$ cp -a temp2 temp3
```

```
$ rm -r temp*
```

Displaying file content

```
$ less proteins.fasta
```

Outputs file content in a paginated fashion

Use arrows [up, down], [enter] and [space] to navigate, press [q] to quit; use /<pattern> for search

```
$ zless proteins.fasta.gz
```

Can also show gzip compressed files

```
$ cat proteins.fasta
```

Outputs the complete file

```
$ head proteins.fasta
```

Outputs the first 10 lines of the file

```
$ tail proteins.fasta
```

Outputs the last 10 lines of the file

Difference between two files

file1.txt

```
one  
two  
three  
four  
five
```

file2.txt

```
one  
two  
  
four  
five  
six
```

```
$ diff file1.txt file2.txt
```

```
3c3  
< three  
---  
>  
5a6  
> six
```

3c3: change line 3 of second file to line 3 of first file

5a6: add line 6 of second file after line 5 of first file

Finding (searching) files

```
$ find . -name "*.txt"
```

```
./file1.txt  
./protein_abundance.txt  
./file2.txt
```

Searching files for content

```
$ grep five *.txt
```

```
file1.txt:five  
file2.txt:five
```

Counting lines in files

```
$ wc -l *.txt
```

```
5 file1.txt  
6 file2.txt  
20489 protein_abundance.txt  
20500 total
```

Redirections to files (<, >, >>)

```
$ echo "Hello world" > poem.txt
```

```
$ echo "Hello universe" >> poem.txt
```

Redirections between commands (|, pipe)

```
$ echo "Hello world" | wc -w
```

```
$ cat protein_abundance.txt | wc -l
```

```
$ grep ">" proteins.fasta | wc -l
```

```
$ grep -c ">" proteins.fasta
```

```
$ grep -v ">" proteins.fasta
```

```
man grep
```

Editing files with "vi"

```
$ vi first_script.sh
```

vi is in command mode
press **i** (enters **insert mode**)

```
#!/bin/bash  
echo "My first script :)"
```

press **ESC** (enters **command mode**)
type **:wq**

```
$ chmod +x first_script.sh  
$ ./first_script.sh
```

set executable flat to first_script.sh
and run the script

```
bash script :)
```

Where to get help?

```
$ man ls
```

```
$ command -h
```

```
$ info ls
```

And after you are done?

```
$ exit
```