

We are 183

Linux & Command Line
and

Life after 183

L23: Monday – Week 14

Reminders

- Grading of core should be back to you soon
- Monday 4/18
 - Reach is Due
- Thursday 4/21
 - Showcase
 - Companies will be there to talk to you

Reminders

- Remember to fill out course evaluations, forward confirmation email to eeecs-evals@umich.edu
 - See directions in email announcement

Discussion Sections

- Tuesday
 - Maxim: Creating a Personal Website
 - Leah: Creating the Perfect Resume
 - Grace: Head Start to 280
 - Anna R.: Resume and Technical Interview
- Wednesday
 - Kevin: Interview Tips and Tricks
- Thursday
 - Michael: Getting Started with Web Development
- Friday
 - Erin: Elevator Speeches and Preparing for Career Fairs

UNIX operating system

- **1969: Project Started at AT&T Bell Labs**
 - by Ken Thompson and Dennis Ritchie
- **1971: First Edition of Unix**
- **1972: Version 2**
 - Rewritten in C (made it portable)
- **1973: Version 5**
 - Licensed for the first time to Universities



UNIX System Allowed:

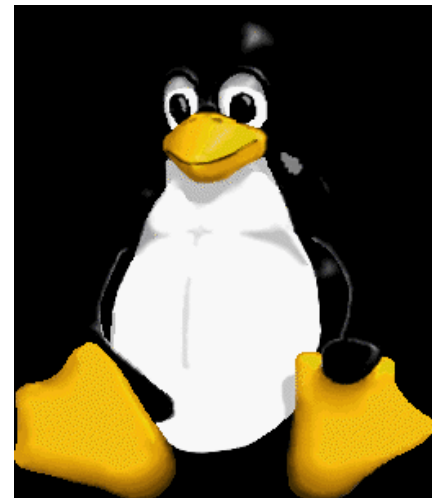
- **Computer aided design**
- **Manufacturing control systems**
- **Laboratory simulations**
- **The Internet**

Without UNIX...

- **The Internet would halt**
- **Most telephone calls could not be made**
- **Electronic commerce would halt**
- **"Jurassic Park" would have never been made**

Linux – A derivative of UNIX

- **Linux is an operating system**
 - 1991: Linus Torvalds created Linux as a hobby
 - Operating system for the autograder
- **Linux Kernel released in 1994**
 - Currently under the GNU General Public License
 - Freely available to everyone
- **Tux is the official mascot**



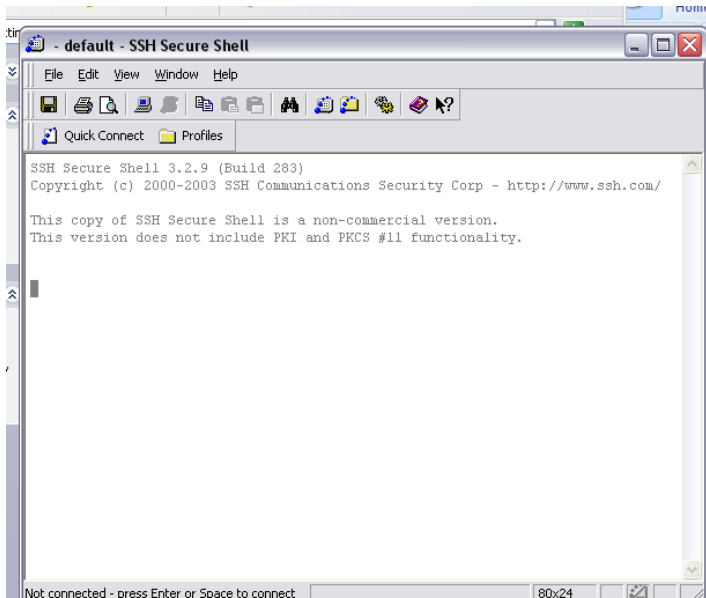
On Mac OS X

- **Mac OS X is based on UNIX**
- **"Terminal Window" is your friend**
 - Gives you "command line" power
 - Linux/Unix command rule

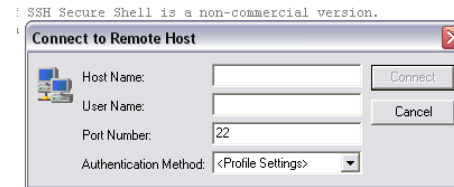
PC Users

- Putty is the recommended command line app
- Download
 - <http://www.itd.umich.edu/bluedisc/>
 - Pick the version for your operating system

PC: Putty Client



Quick Connect



Host Name: login.itd.umich.edu

User Name: uniqname

Connect

Password: your Kerberos passwd

Mac: terminal

```
username$ ssh username@login.itd.umich.edu
```

Comes back and asks for password:

type your Kerberos passwd

Your first linux command: man

■ man ls Manual for "ls"

```
LS(1)                                      BSD General Commands Manual                                      LS(1)
```

NAME

```
ls -- list directory contents
```

SYNOPSIS

```
ls [-ABCFGHLOPRSTUW@abcdefghijklmnopqrstuvwxyz1] [file ...]
```

DESCRIPTION

For each operand that names a file of a type other than directory, **ls** displays its name as well as any requested, associated information. For each operand that names a file of type directory, **ls** displays the names of files contained within that directory, as well as any requested, associated information.

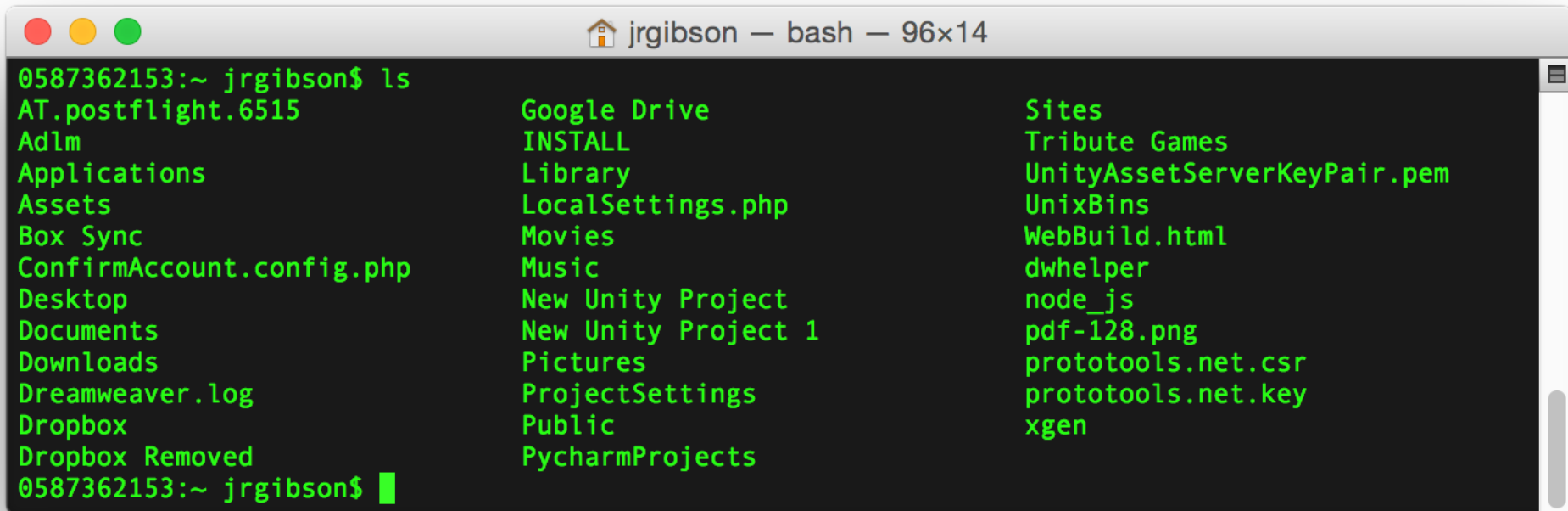
If no operands are given, the contents of the current directory are displayed. If more than one operand is given, non-directory operands are displayed first; directory and non-directory operands are sorted separately and in lexicographical order.

The following options are available:

```
:|
```

List files in directory: ls

- **ls** list files in current directory
“dir” in DOS on PC



A terminal window titled "jrgibson — bash — 96x14" displays the output of the `ls` command. The output is a list of files and directories in the current directory, organized into three columns. The files and directories listed are: AT.postflight.6515, Adlm, Applications, Assets, Box Sync, ConfirmAccount.config.php, Desktop, Documents, Downloads, Dreamweaver.log, Dropbox, Dropbox Removed, Google Drive, INSTALL, Library, LocalSettings.php, Movies, Music, New Unity Project, New Unity Project 1, Pictures, ProjectSettings, Public, PycharmProjects, Sites, Tribute Games, UnityAssetServerKeyPair.pem, UnixBins, WebBuild.html, dwhelper, node_js, pdf-128.png, prototools.net.csr, prototools.net.key, and xgen.

```
0587362153:~ jrgibson$ ls
AT.postflight.6515      Google Drive           Sites
Adlm                   INSTALL               Tribute Games
Applications           Library               UnityAssetServerKeyPair.pem
Assets                 LocalSettings.php     UnixBins
Box Sync               Movies                 WebBuild.html
ConfirmAccount.config.php Music                  dwhelper
Desktop               New Unity Project     node_js
Documents             New Unity Project 1  pdf-128.png
Downloads             Pictures               prototools.net.csr
Dreamweaver.log       ProjectSettings        prototools.net.key
Dropbox               Public                 xgen
Dropbox Removed       PycharmProjects
```

Print Working Directory: **pwd**

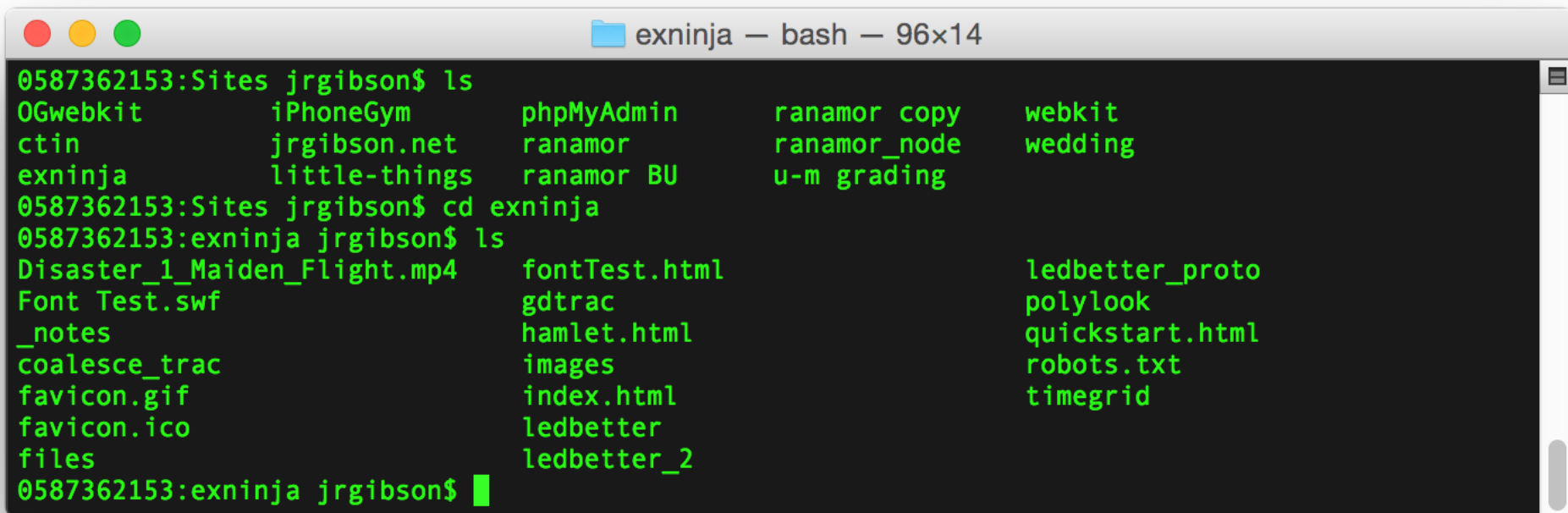
- **pwd** present working directory

```
0587362153:~ jrgibson$ pwd
/Users/jrgibson
0587362153:~ jrgibson$
```


Change Directory: `cd`

■ `cd xxxx` change directory to "xxxx"

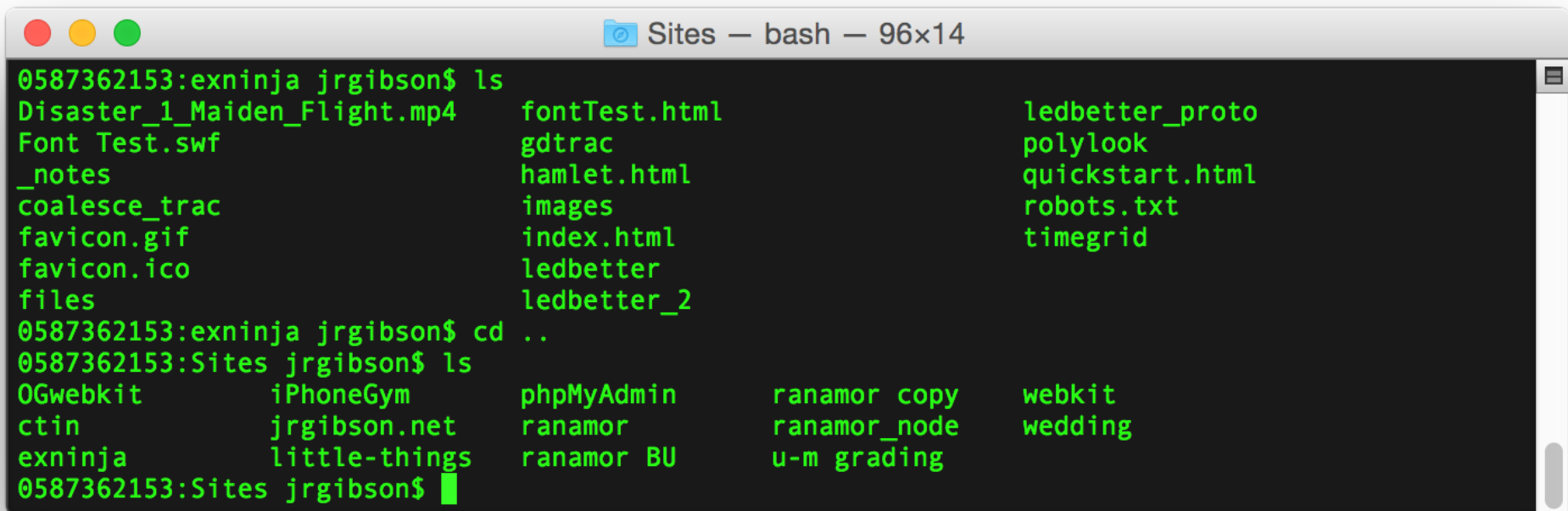
■ Example: `cd exninja`

A terminal window titled "exninja — bash — 96x14" with a blue folder icon. The window shows a sequence of commands and their outputs in green text on a black background. The user starts in the "/Users/jrgibson/Sites" directory and lists files. Then, they use "cd exninja" to move into the "exninja" subdirectory. Finally, they list the files in the new directory. The window has standard macOS window controls (red, yellow, green buttons) in the top-left corner and a scrollbar on the right side.

```
0587362153:Sites jrgibson$ ls
OGwebkit           iPhoneGym          phpMyAdmin          ranamor copy       webkit
ctin               jrgibson.net       ranamor             ranamor_node       wedding
exninja           little-things       ranamor BU          u-m grading
0587362153:Sites jrgibson$ cd exninja
0587362153:exninja jrgibson$ ls
Disaster_1_Maiden_Flight.mp4  fontTest.html      ledbetter_proto
Font Test.swf                gdtrac              polylook
_notes                       hamlet.html         quickstart.html
coalesce_trac                images              robots.txt
favicon.gif                  index.html           timegrid
favicon.ico                  ledbetter
files                        ledbetter_2
0587362153:exninja jrgibson$
```

Change Directory (Up): `cd ..`

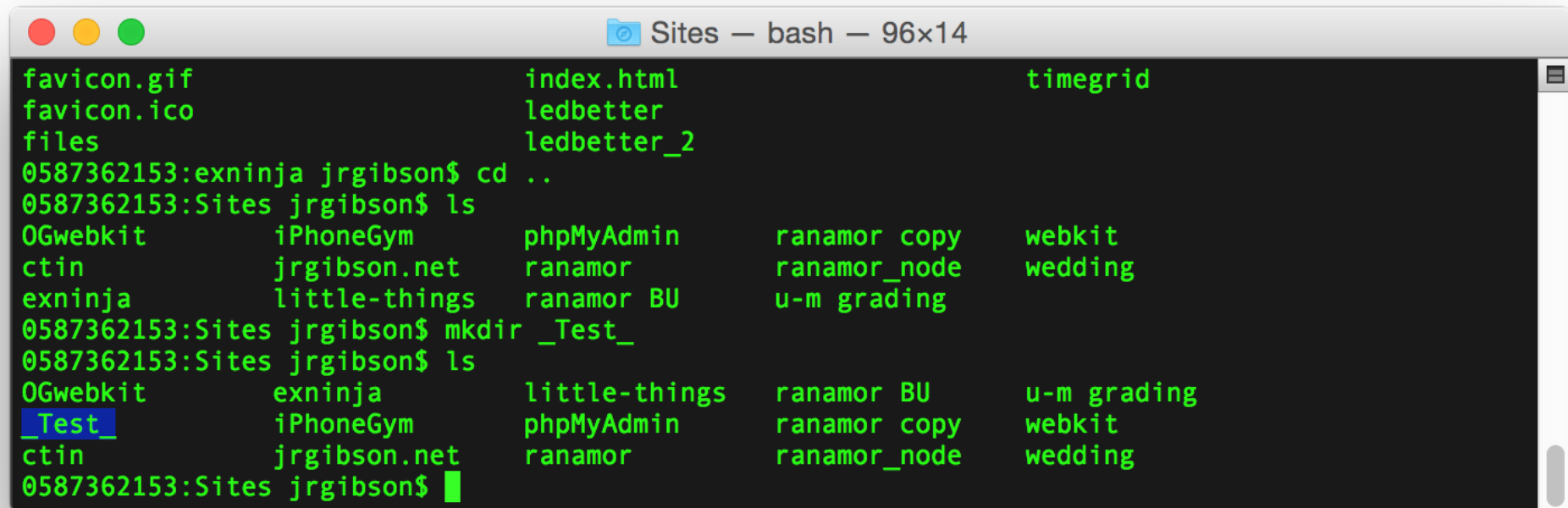
- `cd ..` Goes up one directory
 - Example: `/Sites/exninja/`
`cd ..`
`/Sites/`



```
Sites — bash — 96x14
0587362153:exninja jrgibson$ ls
Disaster_1_Maiden_Flight.mp4  fontTest.html          ledbetter_proto
Font Test.swf                 gdtrac                 polylook
_notes                        hamlet.html            quickstart.html
coalesce_trac                 images                 robots.txt
favicon.gif                   index.html             timegrid
favicon.ico                   ledbetter
files                         ledbetter_2
0587362153:exninja jrgibson$ cd ..
0587362153:Sites jrgibson$ ls
OGwebkit      iPhoneGym      phpMyAdmin      ranamor copy    webkit
ctin           jrgibson.net   ranamor         ranamor_node    wedding
exninja       little-things   ranamor BU      u-m grading
0587362153:Sites jrgibson$
```

Make directory: `mkdir`

- `mkdir xxxx` make directory "xxxx"
 - Example: `mkdir _Test_`
- **Note:** linux does NOT like spaces in names

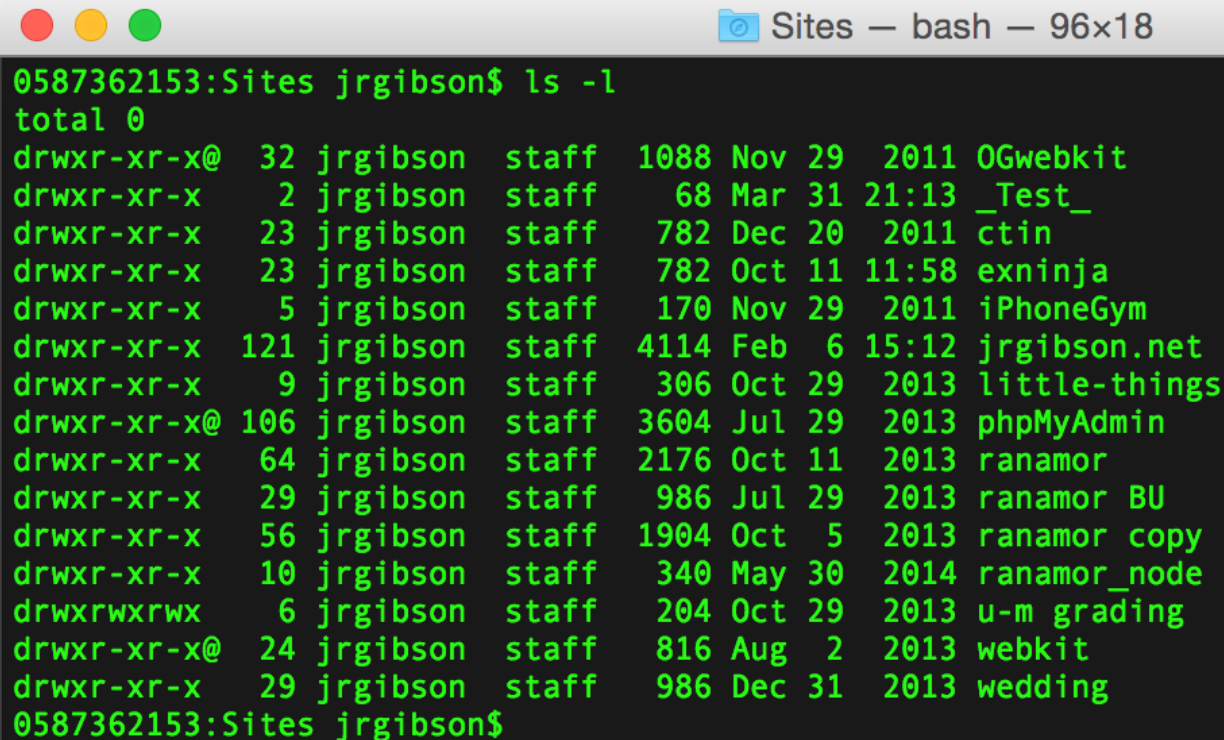


A terminal window titled "Sites — bash — 96x14" showing a series of commands and their outputs. The user is in the "Sites" directory. The initial `ls` command lists various files and directories. Then, the user runs `mkdir _Test_` to create a new directory. A subsequent `ls` command shows the new directory added to the list. The directory name `_Test_` is highlighted in blue in the final output.

```
Sites — bash — 96x14
favicon.gif          index.html          timegrid
favicon.ico          ledbetter
files               ledbetter_2
0587362153:exninja jrgibson$ cd ..
0587362153:Sites jrgibson$ ls
OGwebkit      iPhoneGym      phpMyAdmin      ranamor copy      webkit
ctin          jrgibson.net  ranamor         ranamor_node      wedding
exninja       little-things  ranamor BU      u-m grading
0587362153:Sites jrgibson$ mkdir _Test_
0587362153:Sites jrgibson$ ls
OGwebkit      exninja        little-things  ranamor BU      u-m grading
_Test_        iPhoneGym      phpMyAdmin    ranamor copy    webkit
ctin          jrgibson.net  ranamor       ranamor_node    wedding
0587362153:Sites jrgibson$
```

List files in long form: `ls -l`

- `ls -l` list files in long form



A terminal window titled "Sites — bash — 96x18" showing the output of the `ls -l` command. The output lists 15 files with their permissions, sizes, owners, groups, and timestamps. The files are: OGwebkit, _Test_, ctin, exninja, iPhoneGym, jrgibson.net, little-things, phpMyAdmin, ranamor, ranamor BU, ranamor copy, ranamor_node, u-m grading, webkit, and wedding.

```
0587362153:Sites jrgibson$ ls -l
total 0
drwxr-xr-x@ 32 jrgibson  staff  1088 Nov 29  2011 OGwebkit
drwxr-xr-x  2 jrgibson  staff    68 Mar 31 21:13 _Test_
drwxr-xr-x 23 jrgibson  staff   782 Dec 20  2011 ctin
drwxr-xr-x 23 jrgibson  staff   782 Oct 11 11:58 exninja
drwxr-xr-x  5 jrgibson  staff   170 Nov 29  2011 iPhoneGym
drwxr-xr-x 121 jrgibson  staff  4114 Feb  6 15:12 jrgibson.net
drwxr-xr-x  9 jrgibson  staff   306 Oct 29  2013 little-things
drwxr-xr-x@ 106 jrgibson  staff  3604 Jul 29  2013 phpMyAdmin
drwxr-xr-x 64 jrgibson  staff  2176 Oct 11  2013 ranamor
drwxr-xr-x 29 jrgibson  staff   986 Jul 29  2013 ranamor BU
drwxr-xr-x 56 jrgibson  staff  1904 Oct  5  2013 ranamor copy
drwxr-xr-x 10 jrgibson  staff   340 May 30  2014 ranamor_node
drwxrwxrwx  6 jrgibson  staff   204 Oct 29  2013 u-m grading
drwxr-xr-x@ 24 jrgibson  staff   816 Aug  2  2013 webkit
drwxr-xr-x 29 jrgibson  staff   986 Dec 31  2013 wedding
0587362153:Sites jrgibson$
```

List files in long form: `ls -l`

```
% ls -l
```

```
total 476
```

```
-rwxr-xr-x 1 mdorf staff 455203 Nov 19 2011 a.out
drwxr-xr-x 3 mdorf staff 2048 Jan 21 2004 f02_proj1
drwxr-xr-x 2 mdorf staff 2048 Oct 29 2009 proj4
drwxrwxrwx 3 mdorf staff 2048 Nov 6 2010 test
-rw-r--r-- 1 mdorf staff 455203 Nov 19 2012 conn4.exe
```

List files in long form: `ls -l`

1st Character – File Type

```
% ls -l
```

```
total 476
```

```
-rwxr-xr-x 1 mdorf staff 455203 Nov 19 2011 a.out
drwxr-xr-x 3 mdorf staff 2048 Jan 21 2004 f02_proj1
drwxr-xr-x 2 mdorf staff 2048 Oct 29 2009 proj4
drwxrwxrwx 3 mdorf staff 2048 Nov 6 2010 test
-rw-r--r-- 1 mdorf staff 455203 Nov 19 2012 conn4.exe
```

```
-   normal file
d   directory
s   socket file
l   link file
```

List files in long form: `ls -l`

Permissions

```
% ls -l
```

```
total 476
```

```
-rwxr-xr-x 1 mdorf staff 455203 Nov 19 2011 a.out
drwxr-xr-x 3 mdorf staff 2048 Jan 21 2004 f02_proj1
drwxr-xr-x 2 mdorf staff 2048 Oct 29 2009 proj4
drwxrwxrwx 3 mdorf staff 2048 Nov 6 2010 test
-rw-r--r-- 1 mdorf staff 455203 Nov 19 2012 conn4.exe
```

Next 9 characters

read/write/execution permissions
for

user, group, world
(more later)

List files in long form: `ls -l`

Number of links

```
% ls -l
```

```
total 476
```

<code>-rwxr-xr-x</code>	<code>1</code>	<code>mdorf</code>	<code>staff</code>	<code>455203</code>	<code>Nov 19</code>	<code>2011</code>	<code>a.out</code>
<code>drwxr-xr-x</code>	<code>3</code>	<code>mdorf</code>	<code>staff</code>	<code>2048</code>	<code>Jan 21</code>	<code>2004</code>	<code>f02_proj1</code>
<code>drwxr-xr-x</code>	<code>2</code>	<code>mdorf</code>	<code>staff</code>	<code>2048</code>	<code>Oct 29</code>	<code>2009</code>	<code>proj4</code>
<code>drwxrwxrwx</code>	<code>3</code>	<code>mdorf</code>	<code>staff</code>	<code>2048</code>	<code>Nov 6</code>	<code>2010</code>	<code>test</code>
<code>-rw-r--r--</code>	<code>1</code>	<code>mdorf</code>	<code>staff</code>	<code>455203</code>	<code>Nov 19</code>	<code>2012</code>	<code>conn4.exe</code>

number of links for that file

List files in long form: `ls -l`

Owner

```
% ls -l
```

```
total 476
```

```
-rwxr-xr-x 1 mdorf staff 455203 Nov 19 2011 a.out
drwxr-xr-x 3 mdorf staff 2048 Jan 21 2004 f02_proj1
drwxr-xr-x 2 mdorf staff 2048 Oct 29 2009 proj4
drwxrwxrwx 3 mdorf staff 2048 Nov 6 2010 test
-rw-r--r-- 1 mdorf staff 455203 Nov 19 2012 conn4.exe
```

lists the "Owner" of the files

List files in long form: `ls -l`

Group

```
% ls -l
```

```
total 476
```

```
-rwxr-xr-x 1 mdorf staff 455203 Nov 19 2011 a.out
drwxr-xr-x 3 mdorf staff 2048 Jan 21 2004 f02_proj1
drwxr-xr-x 2 mdorf staff 2048 Oct 29 2009 proj4
drwxrwxrwx 3 mdorf staff 2048 Nov 6 2010 test
-rw-r--r-- 1 mdorf staff 455203 Nov 19 2012 conn4.exe
```

lists which "Group" the files are part of

List files in long form: `ls -l`

Size

```
% ls -l
```

```
total 476
```

<code>-rwxr-xr-x</code>	<code>1</code>	<code>mdorf</code>	<code>staff</code>	<code>455203</code>	<code>Nov 19</code>	<code>2011</code>	<code>a.out</code>
<code>drwxr-xr-x</code>	<code>3</code>	<code>mdorf</code>	<code>staff</code>	<code>2048</code>	<code>Jan 21</code>	<code>2004</code>	<code>f02_proj1</code>
<code>drwxr-xr-x</code>	<code>2</code>	<code>mdorf</code>	<code>staff</code>	<code>2048</code>	<code>Oct 29</code>	<code>2009</code>	<code>proj4</code>
<code>drwxrwxrwx</code>	<code>3</code>	<code>mdorf</code>	<code>staff</code>	<code>2048</code>	<code>Nov 6</code>	<code>2010</code>	<code>test</code>
<code>-rw-r--r--</code>	<code>1</code>	<code>mdorf</code>	<code>staff</code>	<code>455203</code>	<code>Nov 19</code>	<code>2012</code>	<code>conn4.exe</code>

lists the size of the file in
bytes

List files in long form: `ls -l`

Last modified date & time

```
% ls -l
```

```
total 476
```

<code>-rwxr-xr-x</code>	<code>1</code>	<code>mdorf</code>	<code>staff</code>	<code>455203</code>	<code>Nov 19</code>	<code>2011</code>	<code>a.out</code>
<code>drwxr-xr-x</code>	<code>3</code>	<code>mdorf</code>	<code>staff</code>	<code>2048</code>	<code>Jan 21</code>	<code>2004</code>	<code>f02_proj1</code>
<code>drwxr-xr-x</code>	<code>2</code>	<code>mdorf</code>	<code>staff</code>	<code>2048</code>	<code>Oct 29</code>	<code>2009</code>	<code>proj4</code>
<code>drwxrwxrwx</code>	<code>3</code>	<code>mdorf</code>	<code>staff</code>	<code>2048</code>	<code>Nov 6</code>	<code>2010</code>	<code>test</code>
<code>-rw-r--r--</code>	<code>1</code>	<code>mdorf</code>	<code>staff</code>	<code>455203</code>	<code>Nov 19</code>	<code>2012</code>	<code>conn4.exe</code>

lists date file was last
"modified" along with the time if
it was modified recently

List files in long form: `ls -l`

File name

```
% ls -l
```

```
total 476
```

<code>-rwxr-xr-x</code>	<code>1</code>	<code>mdorf</code>	<code>staff</code>	<code>455203</code>	<code>Nov</code>	<code>19</code>	<code>2011</code>	<code>a.out</code>
<code>drwxr-xr-x</code>	<code>3</code>	<code>mdorf</code>	<code>staff</code>	<code>2048</code>	<code>Jan</code>	<code>21</code>	<code>2004</code>	<code>f02_proj1</code>
<code>drwxr-xr-x</code>	<code>2</code>	<code>mdorf</code>	<code>staff</code>	<code>2048</code>	<code>Oct</code>	<code>29</code>	<code>2009</code>	<code>proj4</code>
<code>drwxrwxrwx</code>	<code>3</code>	<code>mdorf</code>	<code>staff</code>	<code>2048</code>	<code>Nov</code>	<code>6</code>	<code>2010</code>	<code>test</code>
<code>-rw-r--r--</code>	<code>1</code>	<code>mdorf</code>	<code>staff</code>	<code>455203</code>	<code>Nov</code>	<code>19</code>	<code>2012</code>	<code>conn4.exe</code>

name of the file

Permissions

d: directory

r: read

w: write

x: execute

```
-rw-r--r-- 1 mdorf staff 455203 Nov 19 2012 conn4.exe
```

Permissions

d: directory

r: read

w: write

x: execute

in groups of threes
owner group others

`-rw-r--r--` 1 mdorf staff 455203 Nov 19 2012 conn4.exe

Permissions

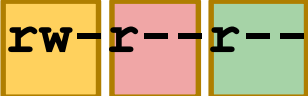
d: directory

r: read

w: write

x: execute

in groups of threes
owner group others

 `-rw-r--r-- 1 mdorf staff 455203 Nov 19 2012 conn4.exe`

'owner'

rw-

'group'

r--

'others'

r--

Permissions

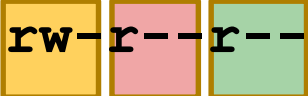
d: directory

r: read

w: write

x: execute

in groups of threes
owner group others

 `-rw-r--r-- 1 mdorf staff 455203 Nov 19 2012 conn4.exe`

'owner'

rw-

'group'

r--

'others'

r--

NO ONE can “execute”

Change permissions: chmod

```
-rw-r--r-- 1 mdorf staff 455203 Nov 19 2012 conn4.exe
```

owner	group	other
-------	-------	-------

"who" are permissions being changed for

u	user who owns the file
---	------------------------

g	group the file belongs to
---	---------------------------

o	other users
---	-------------

a	all of the above
---	------------------

Change permissions: chmod

```
-rw-r--r-- 1 mdorf staff 455203 Nov 19 2012 conn4.exe
```

owner group other

"who" are permissions being changed for

u user who owns the file

g group the file belongs to

o other users

a all of the above

add 'execute'

1) chmod u+x conn4.exe

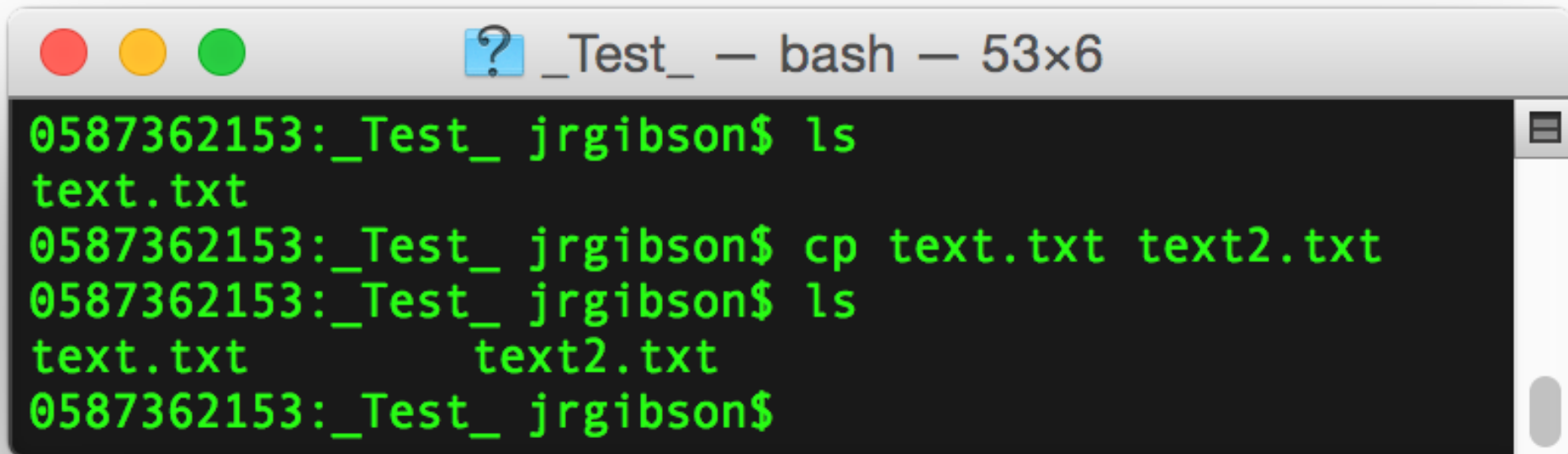
2) chmod g+x conn4.exe

3) chmod o+x conn4.exe

4) chmod a+x conn4.exe

Copy file: **cp**

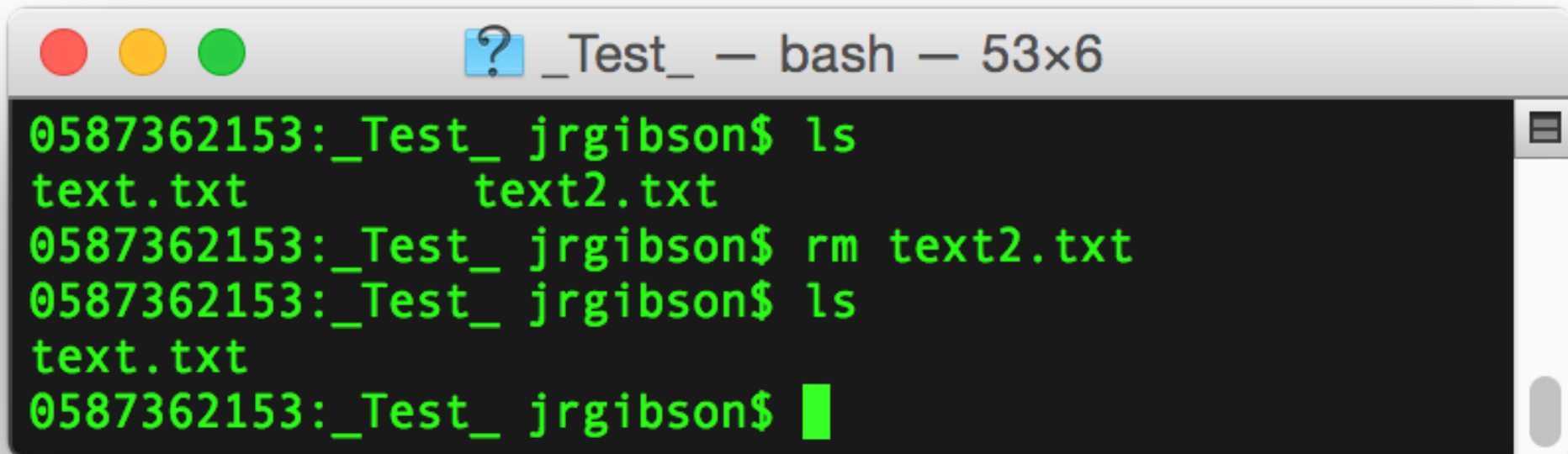
- **cp** xxxx yyyy copy xxxx file to yyyy file
 - Example: **cp test.txt test2.txt**

A terminal window with a title bar containing three colored circles (red, yellow, green) and the text "? _Test_ — bash — 53x6". The terminal has a black background with green text. It shows a sequence of commands and their output: first, 'ls' lists 'text.txt'; second, 'cp text.txt text2.txt' copies the file; third, 'ls' lists both 'text.txt' and 'text2.txt'.

```
0587362153:_Test_ jrgibson$ ls
text.txt
0587362153:_Test_ jrgibson$ cp text.txt text2.txt
0587362153:_Test_ jrgibson$ ls
text.txt      text2.txt
0587362153:_Test_ jrgibson$
```

Remove: **rm**

- **rm** xxxx remove file xxxx
 - Example: **rm test2.txt**

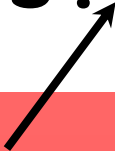
A terminal window with a title bar containing three colored circles (red, yellow, green) and the text "? _Test_ — bash — 53x6". The terminal has a black background with green text. It shows a sequence of commands: 'ls' which lists 'text.txt' and 'text2.txt', followed by 'rm text2.txt', and another 'ls' which now only lists 'text.txt'. The prompt '0587362153:_Test_ jrgibson\$' is repeated for each command.

```
0587362153:_Test_ jrgibson$ ls
text.txt          text2.txt
0587362153:_Test_ jrgibson$ rm text2.txt
0587362153:_Test_ jrgibson$ ls
text.txt
0587362153:_Test_ jrgibson$
```

remove

■ `rm` `fileName`

■ `rm` `proj5.*` `//dangerous`

- 
- This `*` is a "wild card"
 - Anything will match it!
 - ALL of these files would be deleted:
 - `proj5.txt`
 - `proj5.mov`
 - `proj5.exe`
 - `proj5.key`

remove

- `rm fileName`
- `rm proj5.* //dangerous`
- `rm *.* //DANGEROUS`



***.* will match ANY and EVERY file in the directory!!!**

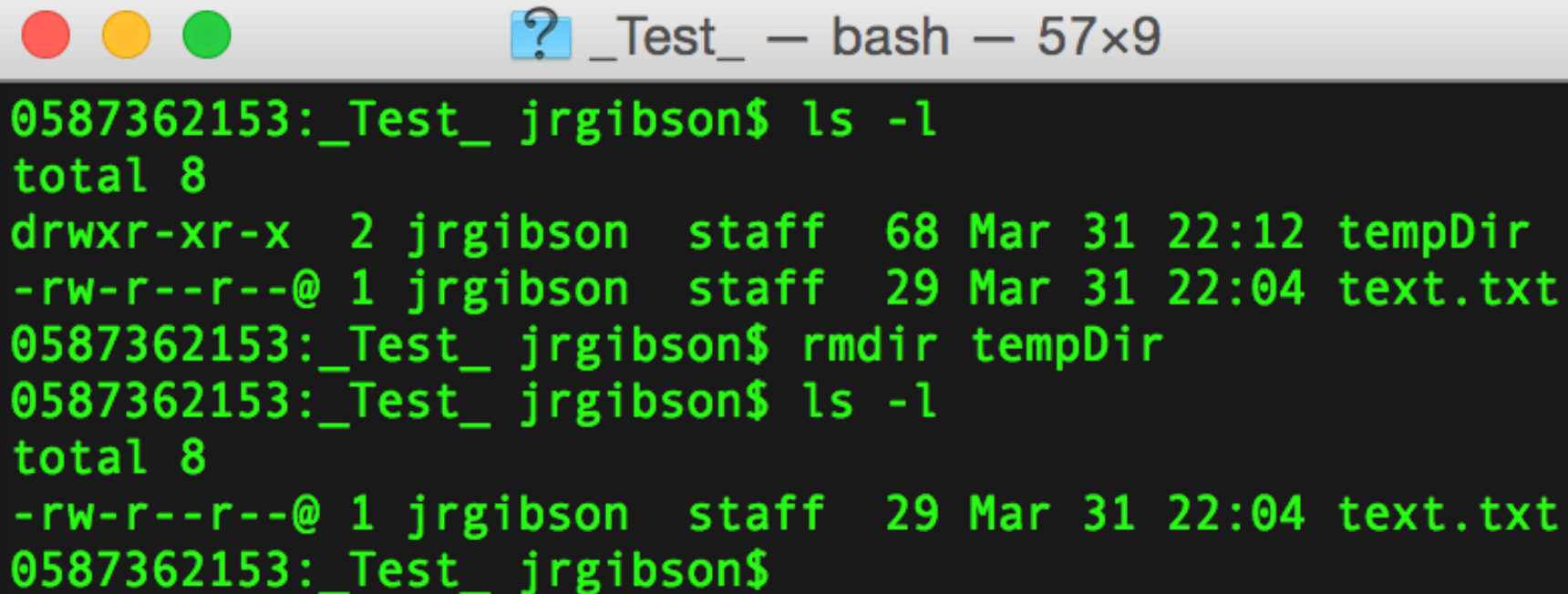
remove

- `rm fileName`
- `rm proj5.* //dangerous`
- `rm *.* //DANGEROUS`
- `rm * //REALLY DANGEROUS`

*** will also match directories in the directory!!!!**

remove directory: **rmdir**

- **rmdir** **dirName** Removes directory **dirName**



A terminal window titled "? _Test_ — bash — 57x9" showing a sequence of commands and their output. The user is in a directory named "_Test_" and is the user "jrgibson". The terminal output shows the initial state with a directory "tempDir" and a file "text.txt". Then, the user runs "rmdir tempDir" to remove the directory. Finally, the user runs "ls -l" again, showing that "tempDir" has been removed, leaving only "text.txt".

```
0587362153:_Test_ jrgibson$ ls -l
total 8
drwxr-xr-x  2 jrgibson  staff   68 Mar 31 22:12 tempDir
-rw-r--r--@ 1 jrgibson  staff   29 Mar 31 22:04 text.txt
0587362153:_Test_ jrgibson$ rmdir tempDir
0587362153:_Test_ jrgibson$ ls -l
total 8
-rw-r--r--@ 1 jrgibson  staff   29 Mar 31 22:04 text.txt
0587362153:_Test_ jrgibson$
```

arrow keys



- use up and down arrow keys to cycle through previously used commands
- easy way to type it once and run it often

compile

```
g++ debug.cpp
```

compile
but don't build executable

```
g++ -c Point.cpp  
g++ -c Line.cpp
```

compile

Compile every *.cpp in current directory and create executable

```
g++ *.cpp
```

compile

Compile every *.cpp in current directory and create executable

```
g++ *.cpp
```

```
./a.out      run the executable
```



references current directory

compile

```
g++ fileName.cpp -o exeName
```

```
./exeName          run the executable
```

redirect input

```
./a.out < data1.txt
```

acts as though input is coming from
keyboard

redirect output

```
./a.out > runOut.txt
```

puts all output to the file

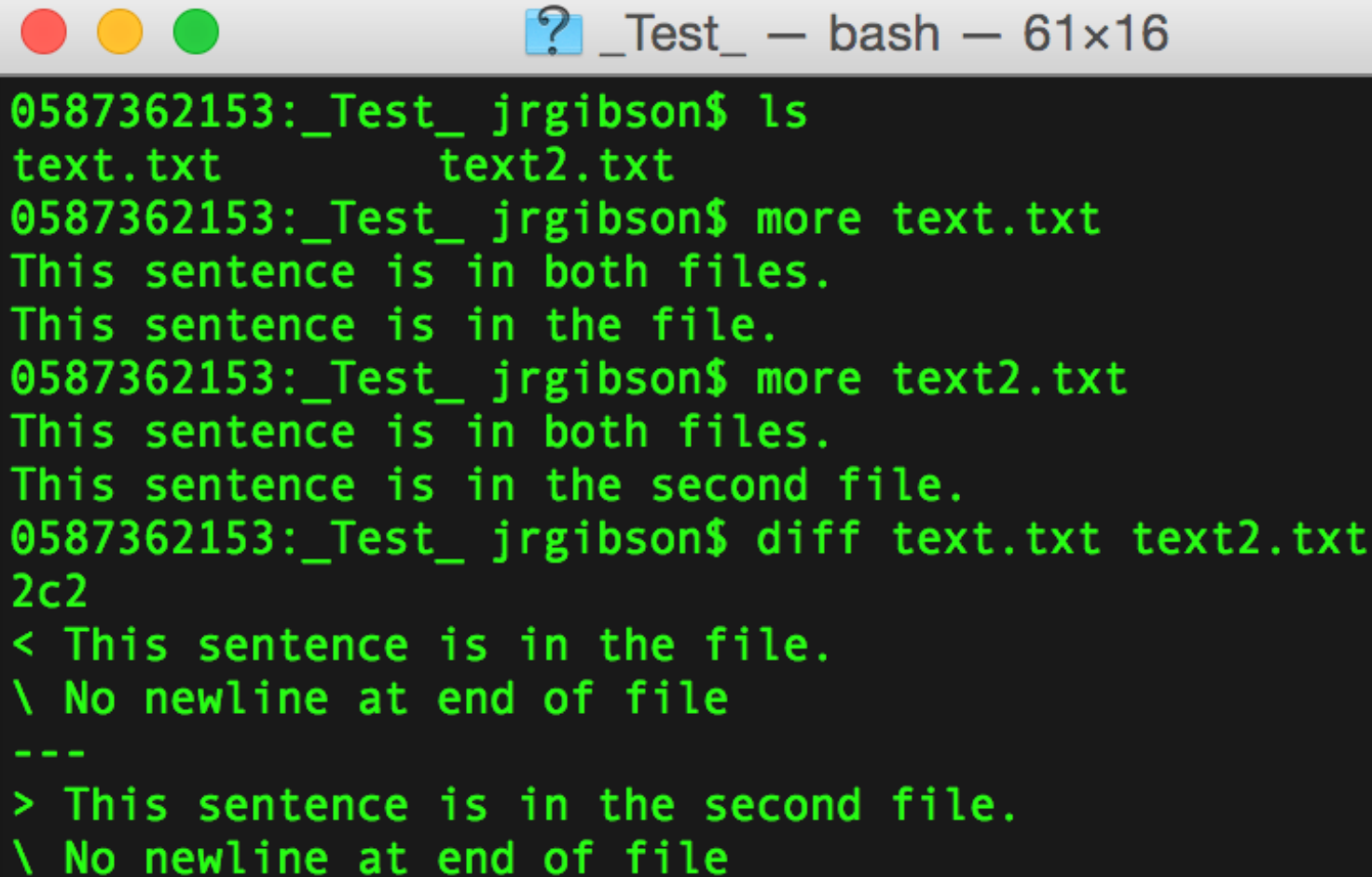
redirect both

```
./a.out < input.txt > runOut.txt
```

reads from “keyboard” and outputs to
the file

Find Differences: `diff`

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

A terminal window titled "_Test_ — bash — 61x16" with standard macOS window controls (red, yellow, green buttons). The terminal shows a user named jrgibson at host 0587362153. They first list files, then view the contents of 'text.txt' and 'text2.txt' using the 'more' command. Finally, they run 'diff text.txt text2.txt', which outputs a comparison showing that 'text.txt' has a line not in 'text2.txt' and 'text2.txt' has a line not in 'text.txt'.

```
0587362153:_Test_ jrgibson$ ls
text.txt      text2.txt
0587362153:_Test_ jrgibson$ more text.txt
This sentence is in both files.
This sentence is in the file.
0587362153:_Test_ jrgibson$ more text2.txt
This sentence is in both files.
This sentence is in the second file.
0587362153:_Test_ jrgibson$ diff text.txt text2.txt
2c2
< This sentence is in the file.
\ No newline at end of file
---
> This sentence is in the second file.
\ No newline at end of file
```

Text editors

- `vim`
- `emacs`
- `Gedit`
- `nano`

Editor: vim



- cheat sheets:

- <http://www.tuxfiles.org/linuxhelp/vimcheat.html>
- <http://bullium.com/support/vim.html>

Minimum you need to know:
:q to exit

FYI:

- develop in Xcode/Visual Studio and upload to a linux system to make sure it works
 - upload is easy
 - PC: Putty (blue disk)
 - Mac: Fugu (blue disk)
- use redirection and different “input files”
- “diff” output with “expected output”



Fugu

Request:

- If any faculty and/or GSI/IA in EECS 280 tells you that you **MUST** develop in linux
- please please relay this to 183 staff
- this is **NOT** true, nor needed
- use what is easy for you
 - upload into linux and rerun your test suite

Life after 183

EECS 280

- Similar to 183 – including workload
- programming in C++
- assumes:
 - conditionals
 - loops
 - functions
 - arrays/vectors
 - continues from there
 - going deeper into classes
 - projects about as complex

EECS 203

- Paper/pencil course
- Learn theory used in EECS 281
 - proof techniques
 - analysis of algorithm
 - probability
 - concepts of trees, graphs

EECS 281

- pre-reqs
 - EECS 203
 - EECS 280
- Workload – higher
 - you design and implement
 - the addition of the "design" is what make it more difficult and time-consuming

Major

- Pre-Reqs

- EECS 203
- EECS 280
- Calc I
- Calc II

need at least a C in all
at least a 2.5 GPA over all

- EECS 281
- EECS 370
- EECS 376
- Stats 250

- 16 hrs of ULCS
 - 4 courses
- Capstone

Minor

- Pre-Reqs
 - EECS 183 need at least a C in all
 - Calc I
 - Taking or having taken one of EECS 203, 280, 281
- EECS 203
- EECS 280
- EECS 281
- One ULCS from a list

We are 183

Wednesday – Security