

• Syllabus Review

→ Kernel Programming

→ Quizzes, nah

→ Weekly HW

→ Kernel Driver: 1) Morse Code generation/encryption } Projects
2) Data Logger→ Raspbian Jessie* HAT: Hardware Attached
on Top• Styling: `if_ () {``function()``sizeof(...)` ← `sizeof` is a macro, no space* K&R → Kern & Richie
invented C (so they're right)• Brush up on pointers→ * is attached to variable on definition `char *pointer;`

• Comments: // is good

• trick compiler w/macro `(do_this(b,c); while(0))`• use the c-library

(Linus thinks he's god)

Kernel Code

Compiling / C

Project I

Networking

SQL

PHP/Python? (web stuff)

Project II

USER SPACE

- CMD LINE
- PERL/Python
- Tools ⇒ text process; msg. system
- RE (regular expressions) ⇒ pattern ~~match~~ matching
useful for removing corrupt lines

Command Line

→ Why a shell?

⇒ Automation (example: printing first-year schedules)

→ shell flavors

⇒ sh (avoid it!) - written by Bourne (program in c)

↓
csh ("c shell"), can do if, case, etc. if...fi (lol nah)

↓

tcsh

sheaff'll use this.

↓
bash ("bourne again shell" lmao)

(we'll use this)

others:

zsh

⋮

* exit (exits program)
("duh")

Bash in the cmd line

↑, recall

ctrl A, begin of line

ctrl C, send term signal (delete text or stop program)

ctrl Z, suspend signal (if terminal is tied up)

ctrl L, redraw screen

history, print past command

⇒ ! 390 (recalls command from line 390 of history)

* bindkey -v (vi keybindings)
-e (emacs)

* env (environment variables)

→ ex.) EDITOR = /usr/bin/vi

man ld.so (loader)

⇒ LD-PRELOAD can load your own lib (w/your own printf or something)

* bash is just a program
→ to set environmental variable:

export LD-LIBRARY-PATH=...

→ search through grep (which searches for reg. expressions)

→ ex. env | grep LIBRARY (pipes connect outs to ins)
std. std.

* ~~vi~~ vi .bashrc → settings
runtime config (maybe, lol)

* putting 'd' after
puts program
in background "bg"
immediately
*"fg" = foreground

grep

grep -E is super useful

// want everytime - make alias for grep

⇒ alias grep 'grep -E' (looks like a loop, but it won't get stuck)

* type 'alias' to see list

* adding new ones just appends to current alias..? (try this)

* output to a file:

env > env.txt (redirects stdout to file env.txt) * alias >> env.txt (appends to file)

* fdisk partitions disks

→ take input from file ⇒ fdisk < input

Quoting

(+csh)
bash

completions
literal string w/sub

!ex.) touch \&

Meta
chars
(change
the
special
meaning)

&

!

~

^

[

]

{

}

?

*

~

[

]

{

}

?

*

~

[

]

{

}

?

*

~

[

]

{

}

?

*

~

[

]

{

}

?

*

~

[

]

{

}

?

*

~

[

]

{

}

?

*

~

[

]

{

}

?

*

~

[

]

{

}

?

*

~

[

]

{

}

?

*

space,
separates
cmd, exec.,
and in args.

! touch !97
results in two files,
cd x

(where 'cd x' was the
output of !97)

fix: touch '!97' ✗

* echo \$USER

↳ accesses env variable

⇒ pi

touch "go home \$USER" ⇒ go home pi

'go " " ' ⇒ go home \$USER

bash completions

grep --, hit double-tabs (shows options)

* ~~grep~~ bash implicitly quotes

* complete // lists completions (a lot)

Globbering: filename matching

* globbing ≠ RE

↓
filename match. for pattern matching

*

^

~

[

]

{

}

?