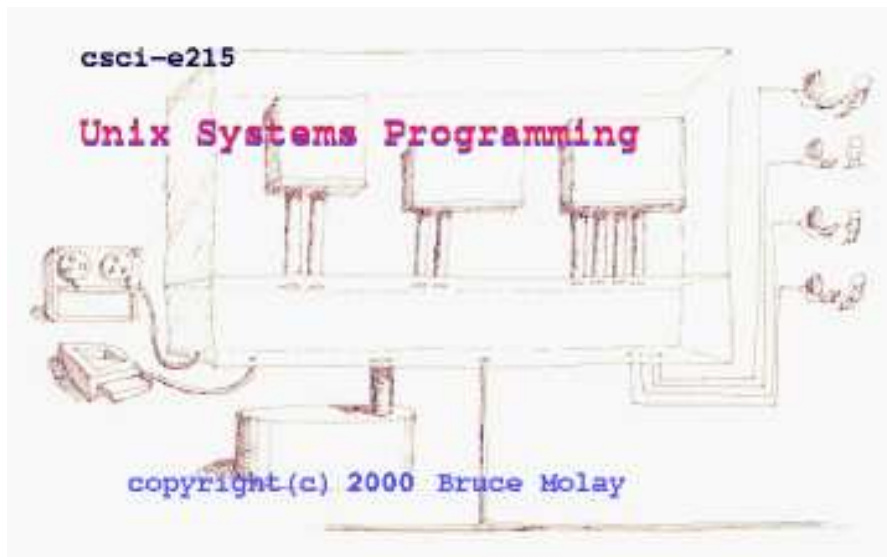


page000.html

Welcome

000



page001.html

001

CSCI-E28
Unix/Linux Systems
Programming

Bruce Molay
molay@fas.harvard.edu

Alexis Layton
Brandon Williams

Outline

Welcome

Where are we going?

How will we get there?

- course mechanics*
- content preview/overview*
- operating environment*
- our approach*

Where are we going?

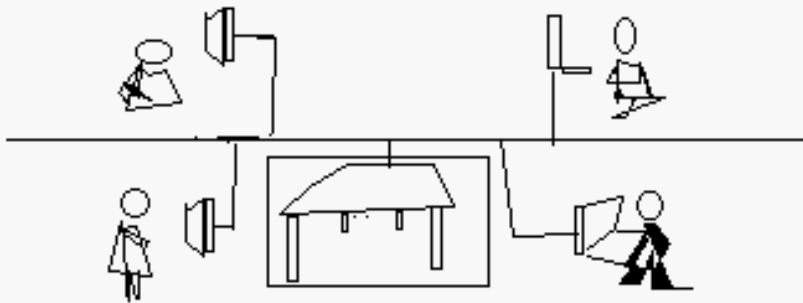
*By the end of the course,
you will have written or be able
to write:*

- a) The Unix tool `bc`*
- b) Internet multi-player bridge*
- c) A web server*

*What skills and ideas do you
need to do these?*

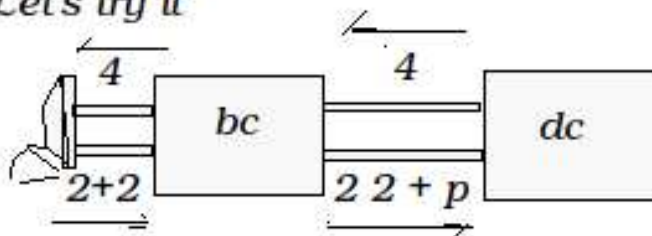
page004.html

004

Internet Bridge*ideas/skills**multiple users**distributed data**communication, cooperation,
coordination*

page005.html

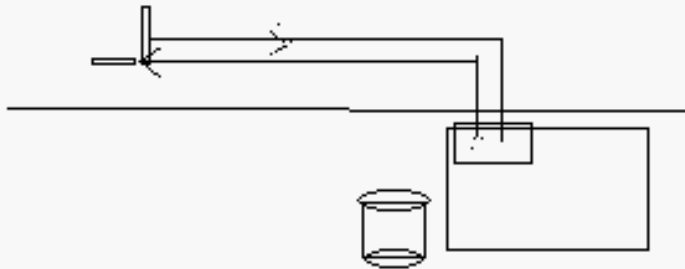
005

Ex2: bc calculator ...*Let's try it**bc is actually a front-end to the dc
calculator**bc translates infix to rev polish notation**Involves: communication,
cooperation,
coordination*

page006.html

006

Ex3: Web server + browser



*multiple programs
communication
cooperation
coordination*

page007.html

007

How do we get there?

Canvas site: Zoom, Ed, Videos

dce-lib215: classes, projects, the rest

we go over the packet of handouts

outline

description

schedule

classes : each week has a page

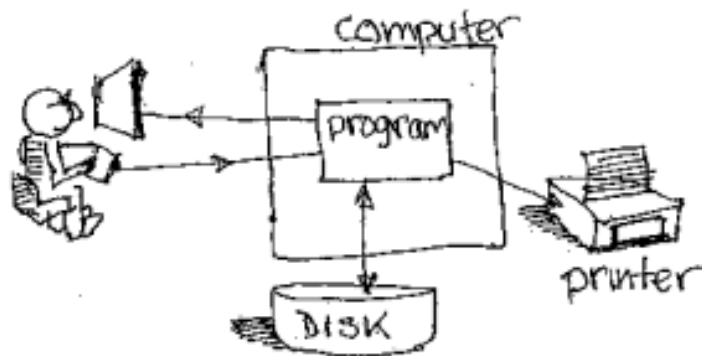
projects page - read the nice documents

required work description/specs

*do student info form so we can schedule
office hours and sections*

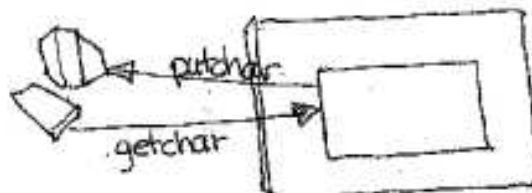
(I) What is systems programming?

① naive, simple program model of programming



typical program in this model

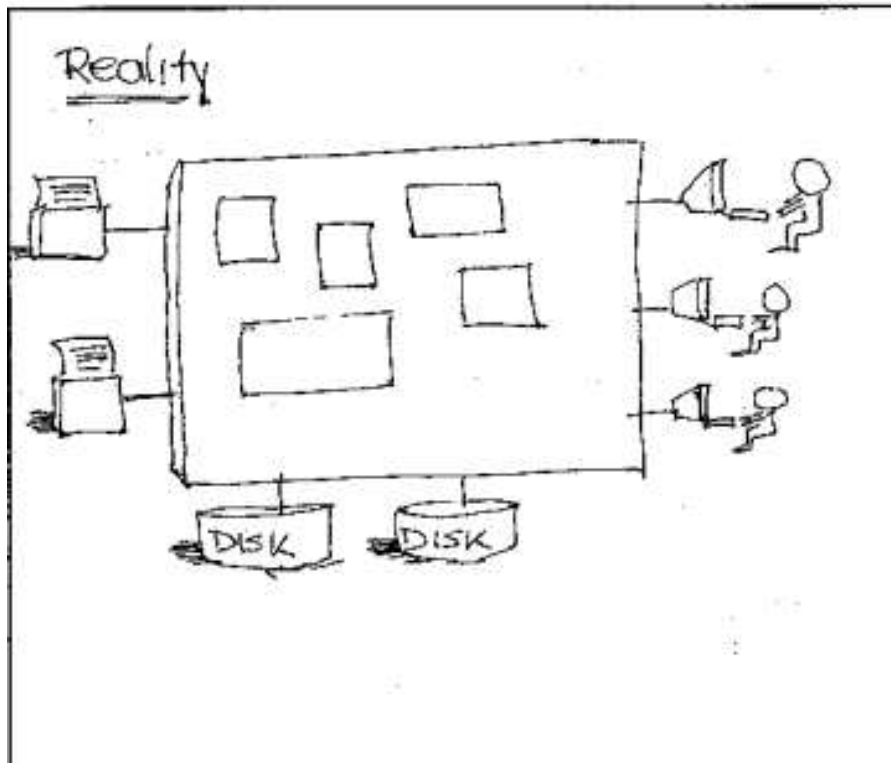
```
main()
{
    int c;
    while ( (c=getchar()) != EOF )
        putchar(c);
}
```



page010.html

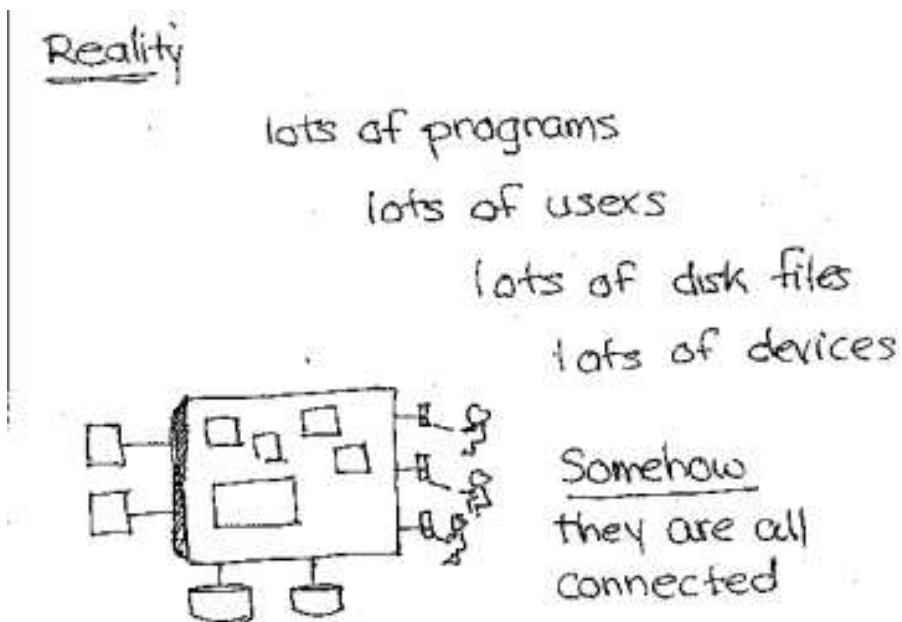
Topics and Approach

010



page011.html

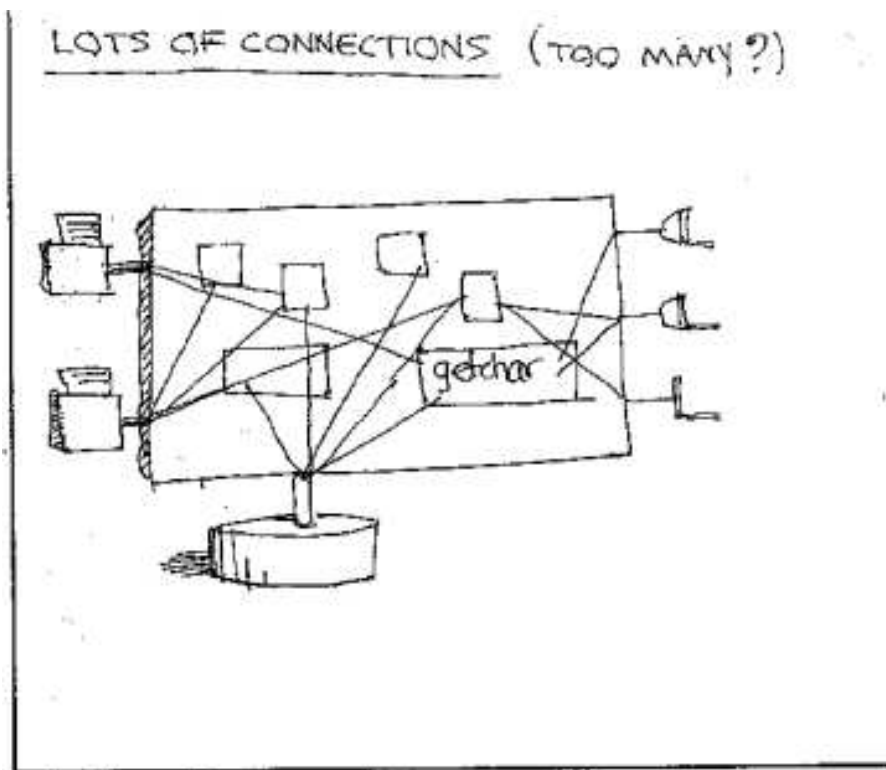
011



page012.html

Using Unix

012

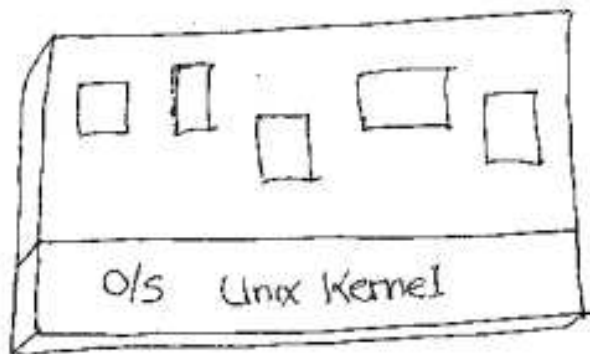


page013.html

013

THE ROLE OF THE OPERATING SYSTEM:

to manage all resources and
connect the various devices
to the correct programs.

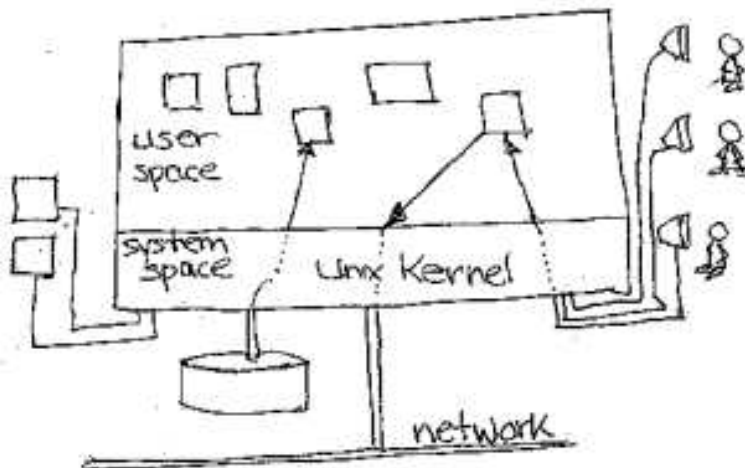


page014.html

014

PROVIDING SERVICES TO PROGRAMS

programs request services from the OS. the OS contains code to provide services.



page015.html

Files and Directories

015

We shall learn to use the services provided by the kernel.

- 1) Input/output
- 2) process management
- 3) memory
- 4) devices: tapes, CDs, mice,...
- 5) timers
- 6) IPC
- 7) network

OUR GOAL

To understand Unix services by:

- ① seeing how they are used
- ② learning about the system calls to invoke the services
- ③ writing our own versions of various systems programs

Unix Systems Programming Tonight's outline

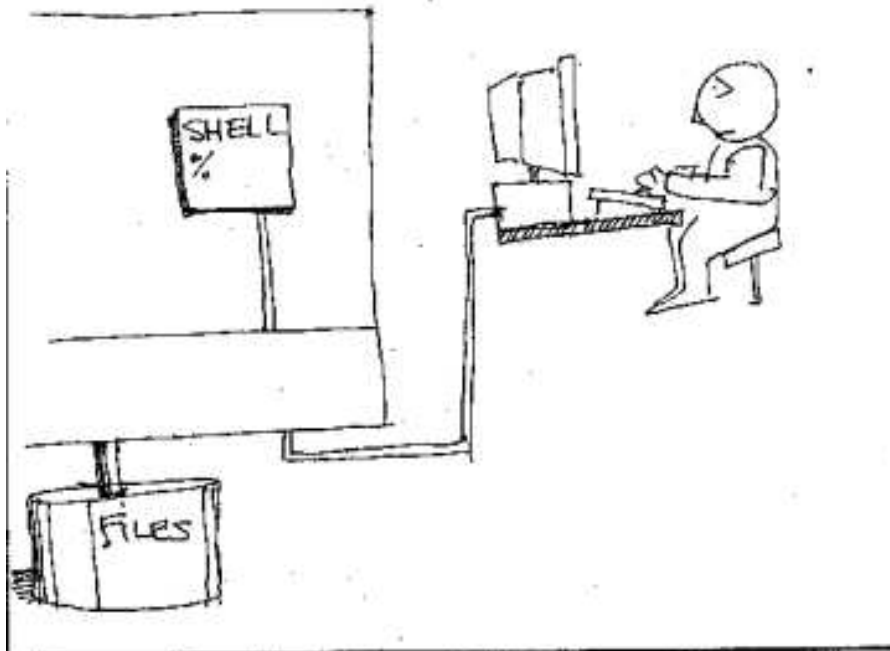
- ~~① course info~~
- ~~② course content~~
- ③ intro to unix
- ④ examples to show what we'll be doing



UNIX FROM THE USER PERSPECTIVE

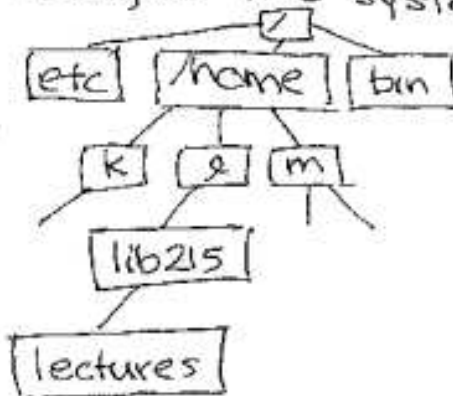
login
files
programs
i/o
pipes
documentation

LOGIN - RUN PROGRAMS - LOGOUT



FILES AND DIRECTORIES

Unix organizes files in a tree-structured directory system and provides commands to view and navigate this system



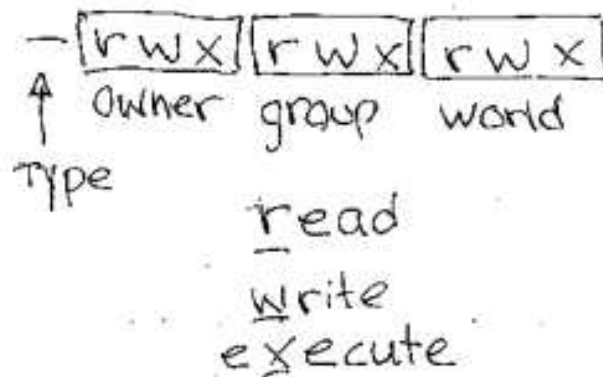
ls
cd
pwd
mkdir
rmdir

Operations on Files

<u>operation</u>	<u>Unix command</u>
examine	cat, more
copy	cp <u>old</u> <u>new</u>
delete	rm
rename	mv <u>old</u> <u>new</u>
print	lpr

FILE PERMISSIONS

Every file has an "owner" and three groups of file permission attributes:



Let's TRY WRITING ONE: more

Usage:

arg(s): process
file content

more filename ...

OR

cmd | more

no args: process
stdin

Let's try writing one: more

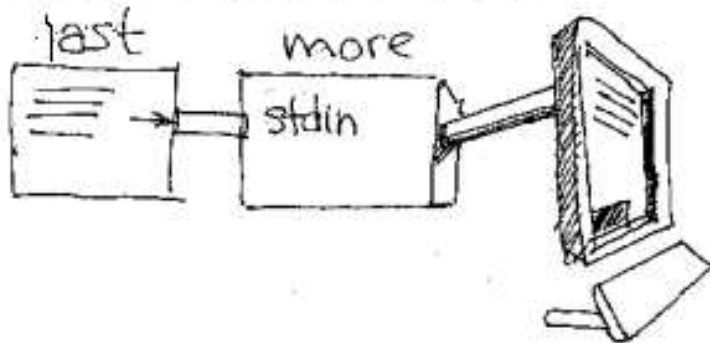
Usage: more filename
or
and|more

Action

→ show 24 lines
→ print MESSAGE
accept ↵, SPACE, Q
if ↵ ADVANCE LINE
IF SPACE
IF Q → exit

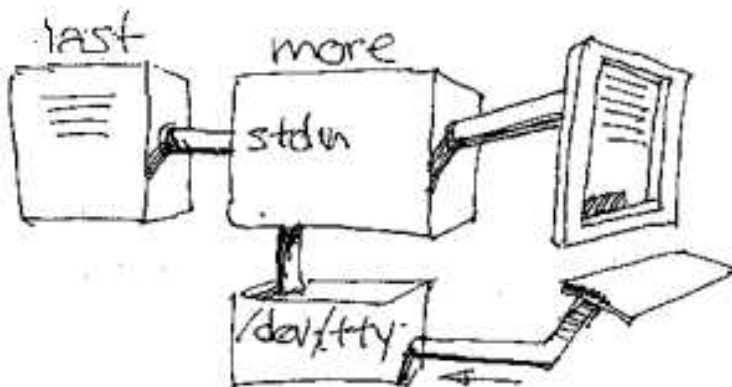
UNIX programming is
not as hard as you
think it is, but it's not
as easy as you
first imagine.

Our first draft

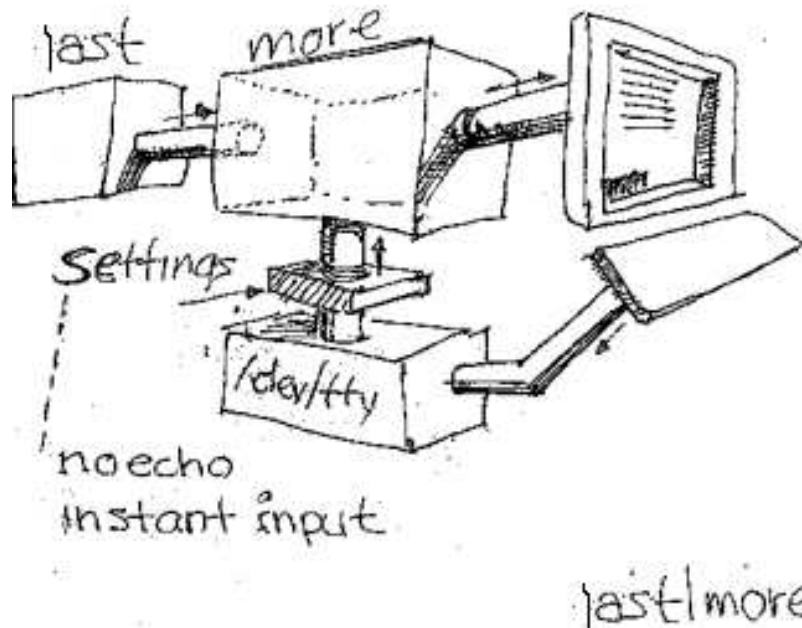


last | more

THE REAL MORE



last | more

THE REAL MORE"more" questions

① % of file?

Ans: count lines?

Ans: the OS knows the
filesize

②

"more" questions

① % of file?

Ans: count lines?

Ans: the OS knows the file size

② REVERSE VIDEO

How to do it correctly?

Ans termcap & terminfo

db of every terminal +
how to control it

