

# Unix II



# Big News (1 of 3)



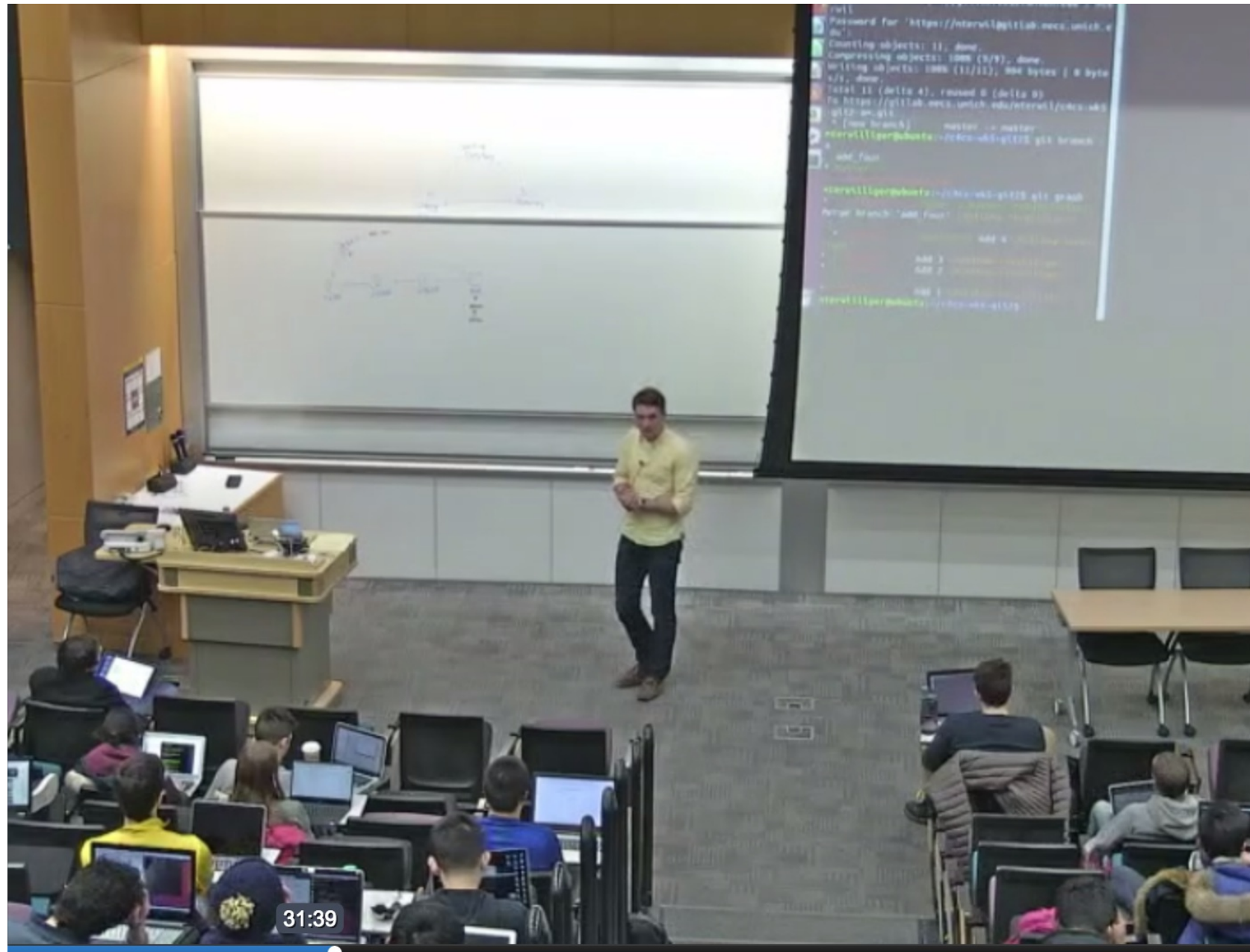
- New edition addition guy... Marcus, Jr.
- Big & Fast

# Big News (2 of 3)



- C4CS Course Creator
- Towner Prize (Outstanding GSI) Winner
- Mr. Chez Betty
- All around Great Person

# Big News (3 of 3)



- Git II Lecturer: IA Matt Terwilliger
- Personal redemption
- #SEEYOUINCLASS

# Administrative

# Regex/Regexp

## What is it?

- 99 Problems
- All the "Big" tools use it
- Try out
  - [www.regular-expressions.info](http://www.regular-expressions.info)
  - [www.regexr.com](http://www.regexr.com)
- Endless amusement during live demos

# Regex/Regexp

## What is it?

- 99 Problems
- All the "Big" tools use it
- Try out
  - [www.regular-expressions.info](http://www.regular-expressions.info)
  - [www.regexr.com](http://www.regexr.com)
- Endless amusement during live demos

## The Basics

- Special Characters `\^$.|?*+()[\{\}`
- Character classes `[<chars>]`
- Shorthand classes `\<char>` and `[[:class:]]`
- Anchors
- Alternation
- Repetition
- Greedy & Lazy
- Grouping
- Backreferencing

# Working on data by line

## Get it out there

- cat
- echo
- head
- tail

## Big Tools

- grep (g/re/p)
- sed
- awk



# Other Tools

- cut
- tee
- tr
- sort
- uniq
- diff
- comm

# 3 Ways to get/use input/output

1. Use pipes (|) to connect multiple commands
2. Use redirection to read/write files
3. Use a subshell \$( ) to assign to variables

# File Redirection

## File descriptors, name and number

- 0: Standard Input (stdin, cin)
- 1: Standard Output (stdout, cout)
- 2: Standard Error (stderr, cerr)

## Operators

- `<` send file as input
- `>` send output to file (create/overwrite)
- `>>` append output to file (create/preserve)
- `>&` merge output
- `<&` merge input
- `<<` "Here Document"

# Scripts

- Just Do It!
  - `history`
- Shebang, Hashbang, #!
- Get that money...
  - `$0` current script
  - `$n` script args 1, 2, 3...
  - `$#` number of args
  - `$*` and `$@` quoted args
  - `$?` the previous return value
- Variables and Arrays
- Math... `$((EXPRESSION))`
- Spacing
  - Critical
  - Maddening

# Control

- Functions

```
# Define
function_name() {
    statements
}

# Invoke
function_name

# Delete
unset .f function_name
```

- Loops

- while (while, do, done)
- for
- until
- select
- break/continue

- Conditionals

- if (if, then, elif, then, else, fi)