Lecture Participation Poll #4

Log onto pollev.com/cse374
Or
Text CSE374 to 22333

Lecture 4: Working with grep and redirects

CSE 374: Intermediate Programming Concepts and Tools

Administrivia

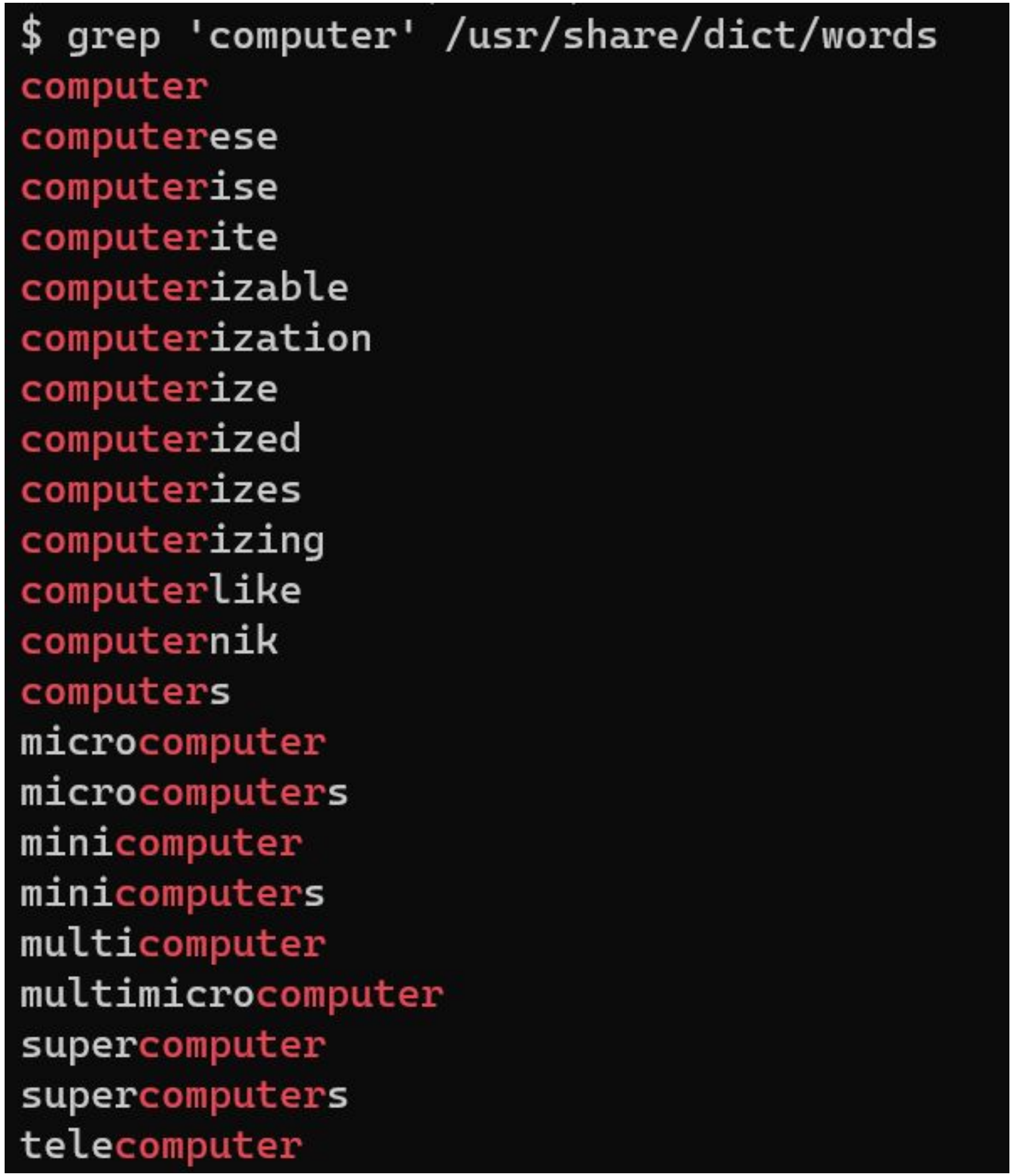
- Course Logistics
  - Office Hours start today – check out #office-hours on discord
    - https://courses.cs.washington.edu/courses/cse374/20au/office-hours/
  - Having issues with klaatu? You can use a VM provided by CSE department
    - https://www.cs.washington.edu/lab/software/linuxhomevm

- Assignments
  - Participation – polls to stay open until next lecture – lectures 1-3 won't count
  - Exercises turn in policy change: for 100% turn in within a week of open, for 90% turn in by midpoint due date 11/6
    - EX1 & EX2 open today and will not be docked any late penalty before 11/6
    - https://courses.cs.washington.edu/courses/cse374/20au/exercises/
  - Homework 1 instructions posted later today – for ec turn in 10/16
    - https://courses.cs.washington.edu/courses/cse374/20au/homeworks/
  - Style guidelines: https://google.github.io/styleguide/shellguide.html

- Stay Connected
  - Don't forget to join the discord https://discord.com/invite/7achjA9
  - Please fill out student survey https://forms.gle/7M7qaHRUAZ4x5tPm8
  - Anonymous Feedback form new location https://feedback.cs.washington.edu/
    - #feedback on discord
  - Want to chat? Make an appointment with Kasey: calendly.com/kasey-champion

grep

- Search for a given string within a given file
  - `grep [options] pattern [files]`
  - EX: `grep “computer” /usr/share/dict/words`
- Helpful Options
  - `-c` : prints count of lines with given pattern
  - `-h` : display matched lines (without filenames)
  - `-i` : ignore case when matching
  - `-l` : display list of filenames with matches



```
$ grep 'computer' /usr/share/dict/words
computer
computerese
computerise
computerite
computerizable
computerization
computerize
computerized
computerizes
computerizing
computerlike
computernik
computers
microcomputer
microcomputers
minicomputer
minicomputers
multicomputer
multimicrocomputer
supercomputer
supercomputers
telecomputer
```

https://www.geeksforgeeks.org/grep-command-in-unixlinux/

Demo: Grep

Redirecting Streams

- Stdout & stderr default to terminal
- Redirect stdout to a file
  - Adding > <file> after a command will overwrite the given file with stdout
    - EX: ./myprogram args < outfile
  - Adding >> <file> after a command will append stdout to the file's existing content
    - EX: ./myprogram args >> outfile
- Stderr can be redirected using 2> and 2>> because 2 represents "stderr"
- If you leave an argument off some utilities they will process input from stdin
  - Ex: grep 'mystring'
    - Will search through stdin
  - EX: foo<bar
    - Runs the program foo, with stdin from file bar
- Read command binds stdin tokens to shell variables
  - EX:
    - Cal > cal.txt
    - Read month year < cal.txt
    - Echo $month of $year

Read doesn't work with | because complicated reasons

Some more commands

Du – disk utilization – prints the disk size of each folder/file passed as args

EX: du myDir test.txt

Common options

-h for human readable (Kilobytes, Megabytes, Gigabytes)

-d 1 limits the depth of the recursion to 1st level of directories

EX: du – h –d 1 .

- Prints the size of all the files/folders in the current directory

Cal – prints current month and year with a text image of the current month

I/O Piping

We can feed the stdout of one process to the stdin of another using a pipe (“|”)

- Data flows from process to the other through multiple transformations seamlessly
- Similar to redirection, but specifically passes streams into other programs instead of their defaults

Example:

- Instead of:
  - du –h –d 1 . > sizes.txt
  - grep ‘M’ sizes.txt
- We can use piping
  - du – h –d 1 . | grep ‘M’

- Piping is effective when you have one set of data that needs to be transformed multiple times
  - Cmd1 | cmd2 – pipe output of cmd1 into input of cmd2

Video: The Magic of Piping

Demo: Stream Redirection

File Permissions

- chmod

Writing your own script files

- Tar
- wget

Lists in Bash

- Lists in bash are strings with multiple words separated by white space
  - Bash does not have arrays