

Pipes & Filters

Module 14

Overview

- In this module:
- Overview of Pipes & Filters
- Viewing the Head or Tail of a File - head & tail
- Finding Data – grep
- Extracting Vertical 'Slices' of a File – cut
- Sorting data - sort
- Removing Duplicates - uniq
- Translating Characters – tr

Pipe

- Pipe

- run two (or more) commands simultaneously

- \$ command1 | command2

- stdout of one is wired to stdin of the next

- the ‘|’ redirects the output of command1 to the input of command2

- very powerful

- Can build complex commands quickly from simple components

Filter

- A program that:
 - reads its standard input
 - makes some change
 - writes to its standard output
- Get to know them
 - This module covers some common ones

Viewing the Head or Tail of a File - head & tail

- head

- show 'head' (first lines) of file

- \$ head -20 long_file

- tail

- show 'tail' (last lines) of file

- \$ tail -3 long_file

- tail -f

- keep file open for reading

Finding Data - grep

- grep
- Filter file for lines containing pattern
 - \$ grep pattern file
- Filter file for lines **not** containing pattern
 - \$ grep -v pattern file

Extracting Vertical 'Slices' of a File - cut

- cut
- extract vertical columns or slices of text
- \$ cut -d: -f1 file

.The sort command

- sort

- Sort lines of text

- \$ sort file

- sort -n

- Sort numerically

- sort -r

- Sort in reverse order

Specifying sort keys

- Sort on field 4 through end of line

- \$ sort -k4 file

- Sort on field 4 and stopping before (not sorting) field 5

- \$ sort -k4,4 file

- Sort on field 6 then numerically in reverse on field 3

- \$ sort -k6,6 -k3,3nr file

The uniq command

- uniq

- Remove repeated lines

- \$ uniq file

- Compares adjacent lines

- Unadjacent lines will not be de-duplicated

- Use with **sort** to operate on whole file

- uniq -c

- Output a count of occurrences

Translating Characters – tr

- tr set1 set2
- translate characters in set1 to corresponding ones in set2
- \$ uptime | tr abc def
- tr reads stdin only, so use pipe or redirect

tr – Specifying Characters

- Individually

- aeiouAEIOU

- Escaped characters

- \n for newline; \t for tab; etc

- Character Ranges

- 0-9; a-zA-Z; etc. Note: no square brackets!

- Character Classes

- [:alnum:] matches all letters and digits

Module 2

- [:alpha:] matches all letters; etc

tr – Complement and Delete

- **tr -c set1 set2**
- **translate characters *not* in set1 to those in set2**
- **tr -d set1**
- **delete any characters which match those in set1**

tr - Squeezing Multiple Occurrences

- `tr -s set1`

- replace multiple occurrences of any character in `set1` with a single occurrence of that character

- `tr -s '\n'`

- Remove multiple newlines