Administrivia

- Probably two more short homeworks, plus project. Information about projects on Web (somewhat preliminary, to be added to soon). Proposals due April 14. Projects due May 12.

Slide 1

"What Command Do I Use To . . . "

- You know about `apropos` as a way to discover new commands. You probably also know that it's not perfect.
- So today (and next time), a tour of some commands I have found useful . . .
- (The point of this tour is not to present details of any of the commands, just to make you aware they exist, so you can follow up on those that seem useful.)

Slide 2

Slide 3

Commands for Working With Text and Other Data

- `script` to capture all terminal input/output. (`exit` to stop capturing.)

  Not as useful as it might be because you also get stuff to control terminal, make colors, etc., but could be a good approach if you need to capture both input and output.

- `strings` to search a file for printable strings.

  Can be useful as a quick-and-dirty (i.e., not necessarily 100% reliable) way of scanning non-text files (e.g., files in MS Office formats) for printable text.

Slide 4

Commands for Working With Text and Other Data, Continued

- `ispell` or `aspell` to check/correct spelling.

- `od` to show data in various forms (binary, hexadecimal, etc.). Useful for finding out exactly what's in a non-text file. Examples:

  `od -t c textfile` to show characters including line-end and other control characters.

  `od -t x1 somefile` to show data a byte at a time in hexadecimal format.

Slide 5

Commands for Printing

- `lpr` to print PostScript or text. (Also seems to print PDF on our machines.) `lpq` to check print queue; `lprm` to cancel a print job.
- `enscript` or `pr` to pretty-print text. Options allow printing in landscape mode with different font sizes, in multi-column format, etc.

Slide 6

Commands to Compress and Archive Data

- `gzip` and `gunzip` to compress/uncompress data. Or try `compress` and `uncompress` (not available on our Linux machines, but found on many UNIX systems).
- `tar` to create UNIX-standard-format "archive" file, a.k.a. "tarball". (Conceptually similar to ZIP archive files — which you can generate, using `zip`.)

  Another way to copy a directory, preserving symbolic links:

  ```
  (cd sourceDir; tar cf - . ) | \
      ( cd target; tar xf - )
  ```

Slide 7

Text-Mode Calculators (`bc` and `dc`)

- Useful in that both support arbitrary precision. (So, if you want to know *exactly* what 2^{100} is . . .)
- I sometimes use from within `vim`, for quick calculations.

Slide 8

(Mostly-)Text-Mode Plotting (`gnuplot`)

- Usually run in graphical mode, but interface is text-only. Help available from within program by typing `help`. (Help is modeled after online help on VAX VMS operating system, and is — different.)
- Can also be run in batch mode — e.g., if you want to be able to easily regenerate plots when data changes.
- Nice for LaTeX users because it can produce output in various LaTeX-friendly formats (including ones that allow final typesetting to use same fonts as document).
- Examples linked from “Sample programs” page (here).

Slide 9

Minute Essay

- Do you have thoughts about a possible project?