CS35L – Fall 2018

Slide set:	5.2
Slide topics:	System call programming
Assignment:	5

Homework 5 (sfrobu.c)

- Rewrite `sfrob` using system calls (`sfrobu`)
- `sfrobu` should behave like `sfrob` except:
  - If stdin is a regular file, it should initially allocate enough memory to hold all data in the file all at once
  - -f option, your program should ignore case while sorting (use the standard toupper function to upper-case each byte after decrypting and before comparing the byte)
- Functions you'll need: `read`, `write`, and `fstat` (read the man pages)
- Measure differences in performance between `sfrob` and `sfrobu` using the `time` command

Homework 5(sfrobs)

- Write a shell script "`sfrobs`" that uses `tr` and the `sort` utility to perform the same overall operation as `sfrobu`
  (support –f option as well)
- Use pipelines (create no temporary files)
- Encrypted input -> tr (decrypt) -> sort (sort decrypted text) -> tr (encrypt) -> encrypted output

Homework 5(report.txt)

- Measure any differences in performance between sfrob and sfrobu using the time command.
- Run your program on inputs of varying numbers of input lines, and estimate the number of comparisons as a function of the number of input lines.
- Use the time command to compare the overall performance of sfrob, sfrobu, sfrobs, sfrobu -f, and sfrobs -f.

Homework 5(report.txt)

- Run your program on inputs of varying numbers of input lines, and estimate the number of comparisons as a function of the number of input lines.
- Varying number of input lines => number of words
- Number of comparisons => keep a counter in the frobcmp() function to check how many times it is being called