**EECS 338**

***Today...***

1. **execvp() + shells**
2. **Parallel computing**
3. **Shared memory**

**Announcements**

* Friday: sockets and pipes (section 3.6)

**Writing your own shell**

* simple-shell.c
* execvp\_demo.c
  + gets(), strtok(), strcmp(), and execvp()
  + <https://www.tutorialspoint.com/c_standard_library/c_function_strtok.htm> (more on Tuesday)
  + <http://www.scs.stanford.edu/nyu/04fa/lab/man/execvp.html>
* Also: Figure 3.9 demonstrates “execlp”
* Alternative to gets(): fgets()

**Parallel computing**

* Getting # of cores:

<http://www.binarytides.com/linux-cpu-information/>

* core\_test\_no\_fork.c: 1 process
* core\_test\_fork.c: 2 processes
* core\_test\_fork4.c: 4 process
* Purpose: looking for speedup

**Shared memory**

* See Figs 3.17 and 3.18
* shm\_producer.c
* shm\_consumer.c
* Use -lrt compiler option for real-time library
  + See makefile
  + [https://docs.oracle.com/cd/E36784 01/html/E36873/librt-3lib.html](https://docs.oracle.com/cd/E36784%2001/html/E36873/librt-3lib.html)