CS 303 Class
Performance monitoring tools.

- Operating system performance monitoring and tuning
    - *top* (table of processes) and *htop* (horizontal table of processes) command. More details: https://serverfault.com/questions/238302/memory-usage-numbers-in-top-htop.
        - # tasks, task states (running, sleeping etc.), CPU % utilization, process information (PID, user, priority, memory: virtual, resident (code + data)), state.
    - *ps* (process snapshot) command. More details:http://man7.org/linux/man-pages/man1/ps.1.html and at https://www.tecmint.com/ps-command-examples-for-linux-process-monitoring/
        - ps: running processes in the current shell.
        - ps - A: every active process in the system.
        - ps -ef: full format listing.
        - ps -X: user running processes.
        - ps -fp pid: list process by PID.
        - ps -e --forest: processes in a tree format.
    - *vmstat* command. More at: https://www.tecmint.com/linux-performance-monitoring-with-vmstat-and-iostat-commands/
    - https://opensource.com/article/17/11/bccbpf-performance
        - vmstat -a: active and inactive memory, swapped in (si), swapped out (so).
        - vmstat -s: print various counters.
    - *iostat* and *netstat* commands.
    - https://www.tecmint.com/20-netstat-commands-for-linux-network-management/
        - netstat -s
        - netstat -su
    - *strace* command. https://www.tecmint.com/strace-commands-for-troubleshooting-and-debugging-linux/
        - strace df -h: traces all system calls made by command df (this command reports file system disk space usage).
        - sudo strace -p PID: trace by PID.
        - sudo strace -c -p PID: summary by PID (need to d Ctrl + c to get this).
    - *bcc* (Berkeley compiler collection) tools. Show some screenshots/slides. https://opensource.com/article/17/11/bccbpf-performance