

## Task Checklist

Name: CPU usage tracker

### Functionality:

- ☒ Thread **Reader**
- ☒ Thread **Analyzer**
- ☒ Thread **Printer**
- ☐ Thread **Watchdog** (have some thoughts on how I would implement)
- ☐ Thread **Logger**
- ☒ **SIGTERM** handler

### Requirements:

- ☒ Modern C (**C99** or higher)
- ☒ Build system: **Makefile**
- ☒ Build system supports both *gcc* and *clang* compile modes
  - Compiled with **no warnings**
    - ☒ gcc
    - ☐ clang (has some non critical warnings)
- ☒ **Git** (1 functionality = 1 described commit)
- ☒ Use **Valgrind** (to deal with memory leaks)
- ☒ Program **doesn't have any memory leak**
- ☐ At least 1 automatic test (could be unit test or any other)
- ☐ Application properly works on any Linux distributions
  - ☒ **tested on Ubuntu and Debian**
- ☒ Understand and use **procfs** file system to properly read data.
- ☒ Use **global variable or structure** to **send data between** threads.
- ☒ Use "**Consumer-Producer Problem**" to send data between threads.
- ☒ Implement **data buffering** (RingBuffer or Queue).
- ☒ Use example to implement **signal handler**.
- ☒ For concurrency **use pthreads** or C11 feature.
- ☐ **Use assert** for unit tests.
- ☒ Split app into modules which can be tested.
- ☐ Test each module.
- ☒ Try use OOP principles like KISS, DRY, SOLID.