Lab Exercise

- ***Cron* daemon**
  - Standard tool for running commands on a predetermined schedule
  - Automatically start when the system boots
  - Cron configuration file (*crontab*)
    - » List of commands and their invocation times
    - » *Cron* invokes commands at the predefined times

- **Make your simple *cron* daemon**

Lab Exercise

- **Configuration file**
  - Same path as *cron* daemon execution
  - Format
    - » *minute(0~59) hour(0~23) executable_file*
  - Three arguments are separated by whitespace
    - » e.g.,

```
sw@SW:~/swe2024/week6/cron$ cat crontab
* * /home/sw/swe2024/week6/cron/hello.sh
```

  - Rule matching
    - » * matches everything
    - » Specific number matches minute or hour exactly
    - » Ex) * * hello.sh -> executes hello.sh, every minute
      3 * hello.sh -> executes hello.sh, 3rd minute, every hour
      5 4 hello.sh -> executes hello.sh, 5th minute, 4 AM

Lab Exercise

- **Example**
  1. Configure the ./crontab file
     - » Ex)

```
sw@SW:~/swe2024/week6/cron$ cat crontab
* * /home/sw/swe2024/week6/cron/hello.sh
```

```
sw@SW:~/swe2024/week6/cron$ cat /home/sw/swe2024/week6/cron/hello.sh
echo "Hello world" >> /tmp/hello.txt
```

     - » Give execute permission (*chmod +x hello.sh*)
  2. Execute simple *cron* daemon
  3. Terminate a *cron* daemon using kill command
     - » *kill -9 <pid>* : terminate a process using process_id

```
sw          7558    1266  0 18:00 ?        00:00:00 ./cron
sw          7561    6514  0 18:00 pts/0    00:00:00 ps -ef
sw@SW:~/swe2024/week6/cron$ kill -9 7558
```

Lab Exercise

- **Make simple *cron* daemon**
  - Download skeleton code
    - » https://drive.google.com/file/d/1GG_a22GUAuUqT5ZjyA0IgFDTPTY2EjQk/view?usp=sharing
  - You should use **struct tm *tm**
    - » tm->tm_min: current minute
    - » tm->tm_hour: current hour
  - You should sleep until the next job is due to be run
  - Useful API
    - » *int atoi*(*const char *ptr*): convert a string to an integer
    - » *unsigned int sleep*(*unsigned int seconds*): sleep for a specified number of seconds

Exercise Hint

- strtok_r function

```
char str[] = "System Software Laboratory";
char *token;
char *pos = str;

while ((token = strtok_r(pos, " ", &pos)))
       printf("%s\n", token);
```

Lab Exercise

- **Submit your lab exercise**
  - via iCampus
  - Collect your **source codes** and **Makefile** into **tar.gz** format

    $ tar cvzf *student_id.tar.gz* all_your_files
  - We'll grade your submission with **make**
    - » If compilation fails, your points for this exercise will be zero

- **Write your questions in the iCampus Q&A board**

- **How to move files between host OS and guest OS**
  - Use **Drag n Drop**, **Shared Clipboard** or **Shared folder**
  - https://www.tecmint.com/install-virtualbox-guest-additions-in-ubuntu/