

Soal Shift Modul 2
Sistem Operasi 2020

1. Waktu pengerjaan dimulai senin(9/3) pukul 13.00 WIB hingga jumat(13/3) pukul 18.00 WIB.
2. Praktikan diharapkan membuat laporan penjelasan dan penyelesaian soal dalam bentuk Readme(github).
3. Format nama repository github "SoalShiftSISOP20_modul2_xyy"
(contoh:SoalShiftSISOP20_modul2_A01).
4. Struktur repository seperti berikut:
 SoalShiftSISOP20_modul2_xyy:
 ---soal1:
 ---soal1.c
 ---soal2:
 ---soal2.c
 ---soal3:
 ---soal3.c
5. Setelah pengerjaan selesai, semua program C ditaruh di github masing - masing kelompok, dan link github diletakkan pada form yang disediakan.
6. Commit terakhir maksimal 10 menit setelah waktu pengerjaan berakhir. Jika melewati maka akan dinilai berdasarkan commit terakhir.
7. Jika tidak ada pengumuman perubahan soal oleh asisten, maka soal dianggap dapat diselesaikan.
8. Jika ditemukan soal yang tidak dapat diselesaikan, harap menuliskannya pada Readme beserta permasalahan yang ditemukan.
9. Praktikan tidak diperbolehkan menanyakan jawaban dari soal yang diberikan kepada asisten maupun praktikan dari kelompok lainnya.
- 10. Jika ditemukan indikasi kecurangan dalam bentuk apapun di pengerjaan soal shift, maka nilai dianggap 0 (DIBACA: **NOL**).**
11. Pengerjaan soal shift sesuai dengan modul yang telah diajarkan.
12. Tidak boleh menggunakan fungsi system() dan shell script(ex: `execve("/bin/bash", ["curang.sh", NULL]);`).

1. Buatlah program C yang menyerupai crontab untuk menjalankan script bash dengan ketentuan sebagai berikut:
 - a. Program menerima 4 argumen berupa:
 - i. Detik: 0-59 atau * (any value)
 - ii. Menit: 0-59 atau * (any value)
 - iii. Jam: 0-23 atau * (any value)
 - iv. Path file .sh
 - b. Program akan mengeluarkan pesan error jika argumen yang diberikan tidak sesuai
 - c. Program hanya menerima 1 config cron
 - d. Program berjalan di background (daemon)
 - e. Tidak boleh menggunakan fungsi system()

Contoh: `./program * 34 7 /home/somi/test.sh`

Program dengan argumen seperti contoh di atas akan menjalankan script test.sh setiap detik pada jam 07:34.

2. Shisoppu mantappu! itulah yang selalu dikatakan Kiwa setiap hari karena sekarang dia merasa sudah jago materi sisop. Karena merasa jago, suatu hari Kiwa iseng membuat sebuah program.
 - a. Pertama-tama, Kiwa membuat sebuah folder khusus, di dalamnya dia membuat sebuah program C yang per 30 detik membuat sebuah folder dengan nama timestamp [YYYY-mm-dd_HH:ii:ss].
 - b. Tiap-tiap folder lalu diisi dengan 20 gambar yang di download dari <https://picsum.photos/>, dimana tiap gambar di download setiap 5 detik. Tiap gambar berbentuk persegi dengan ukuran $(t\%1000)+100$ piksel dimana t adalah detik Epoch Unix. Gambar tersebut diberi nama dengan format timestamp [YYYY-mm-dd_HH:ii:ss].
 - c. Agar rapi, setelah sebuah folder telah terisi oleh 20 gambar, folder akan di zip dan folder akan di delete(sehingga hanya menyisakan .zip).
 - d. Karena takut program tersebut lepas kendali, Kiwa ingin program tersebut mengenerate sebuah program "killer" **yang siap di run(executable)** untuk menterminasi semua operasi program tersebut. Setelah di run, program yang menterminasi ini lalu akan mendelete dirinya sendiri.
 - e. Kiwa menambahkan bahwa program **utama** bisa dirun dalam dua mode, yaitu MODE_A dan MODE_B. untuk mengaktifkan MODE_A, program harus dijalankan dengan argumen -a. Untuk MODE_B, program harus dijalankan dengan argumen -b. Ketika dijalankan dalam MODE_A, program utama akan langsung menghentikan semua operasinya ketika program killer dijalankan. Untuk MODE_B, ketika program killer dijalankan, program utama akan berhenti tapi membiarkan proses di setiap folder yang masih berjalan sampai selesai(semua folder terisi gambar, terzip lalu di delete).

Kiwa lalu terbangun dan sedih karena menyadari programnya hanya sebuah mimpi. Buatlah program dalam mimpi Kiwa jadi nyata!

Catatan:

- Tidak boleh memakai `system()`.
- Program utama harus *ter-detach* dari terminal

Hint:

- Gunakan fitur `picsum.photos` untuk mendapatkan gambar dengan ukuran tertentu
- Epoch Unix bisa didapatkan dari `time()`

3. Jaya adalah seorang programmer handal mahasiswa informatika. Suatu hari dia memperoleh tugas yang banyak dan berbeda tetapi harus dikerjakan secara bersamaan (multiprocessing).

- Program buatan jaya harus bisa membuat dua direktori di `"/home/[USER]/modul2/"`. Direktori yang pertama diberi nama `"indomie"`, lalu lima detik kemudian membuat direktori yang kedua bernama `"sedaap"`.
- Kemudian program tersebut harus meng-ekstrak file `jpg.zip` di direktori `"/home/[USER]/modul2/"`. Setelah tugas sebelumnya selesai, ternyata tidak hanya itu tugasnya.
- Diberilah tugas baru yaitu setelah di ekstrak, hasil dari ekstrak tersebut (di dalam direktori `"/home/[USER]/modul2/jpg/"`) harus dipindahkan sesuai dengan pengelompokan, semua file harus dipindahkan ke `"/home/[USER]/modul2/sedaap/"` dan semua direktori harus dipindahkan ke `"/home/[USER]/modul2/indomie/"`.
- Untuk setiap direktori yang dipindahkan ke `"/home/[USER]/modul2/indomie/"` harus membuat dua file kosong. File yang pertama diberi nama `"coba1.txt"`, lalu 3 detik kemudian membuat file bernama `"coba2.txt"`.
(contoh : `"/home/[USER]/modul2/indomie/{nama_folder}/coba1.txt"`).

Karena Jaya terlalu banyak tugas dia jadi stress, jadi bantulah Jaya agar bisa membuat program tersebut.

Catatan :

- Tidak boleh memakai `system()`.
- Tidak boleh memakai function C `mkdir()` ataupun `rename()`.
- Gunakan `exec` dan `fork`
- Direktori `."` dan `.."` tidak termasuk