**Przedmiot:** Systemy operacyjne - laboratorium

**Tematyka:** Procesy

**Informacje wstępne**

W ramach przygotowań do laboratorium, proszę zapoznać się z:

* działaniem programów ps, top, htop, pstree
* lokalizacją plików urządzenia pamięci:
  + $ cat /proc/meminfo,
  + $ ls -alh /proc/kcore
* Sprawdzić położenie pamięci swap:
  + cat /etc/fstab | grep swap
  + sudo fdisk -l /dev/sda
* działaniem programów jak free
* man 5 proc
* zawartością /proc/[pid]/
  + /proc/[pid]/cmdline Nazwa polecenia za pomocą którego utworzono proces
  + /proc/[pid]/io Statystyki operacji wejścia wyjścia
  + /proc/[pid]/status Status procesu, wykorzystanie pamięci
  + /proc/[pid]/limits Limity zasobów
  + /proc/[pid]/sched Informacje dotyczące szeregowania procesu
  + /proc/[pid]/fdinfo/ Informacje o plikach które proces otworzył
  + /proc/[pid]/fd/ Deskryptory plików które proces otworzył
  + /proc/[pid]/net/ Informacje dotyczące sieci
* Przykład:
  + Uruchomić: xclock
  + Uruchomić ps aux | grep xclock lub: pidof xclock
  + Oglądnąć w /proc
* Proszę zapoznać się z działanie funkcji *fork()* języka C (np.: man fork).

$ xclock - w jednym terminalu

$ pidof xclock - to w drugim i odczytujemy numer PID procesu xclock

$ cat /proc/[tutaj ten numer bez nawiasów]/cmdline - to też w tym drugim

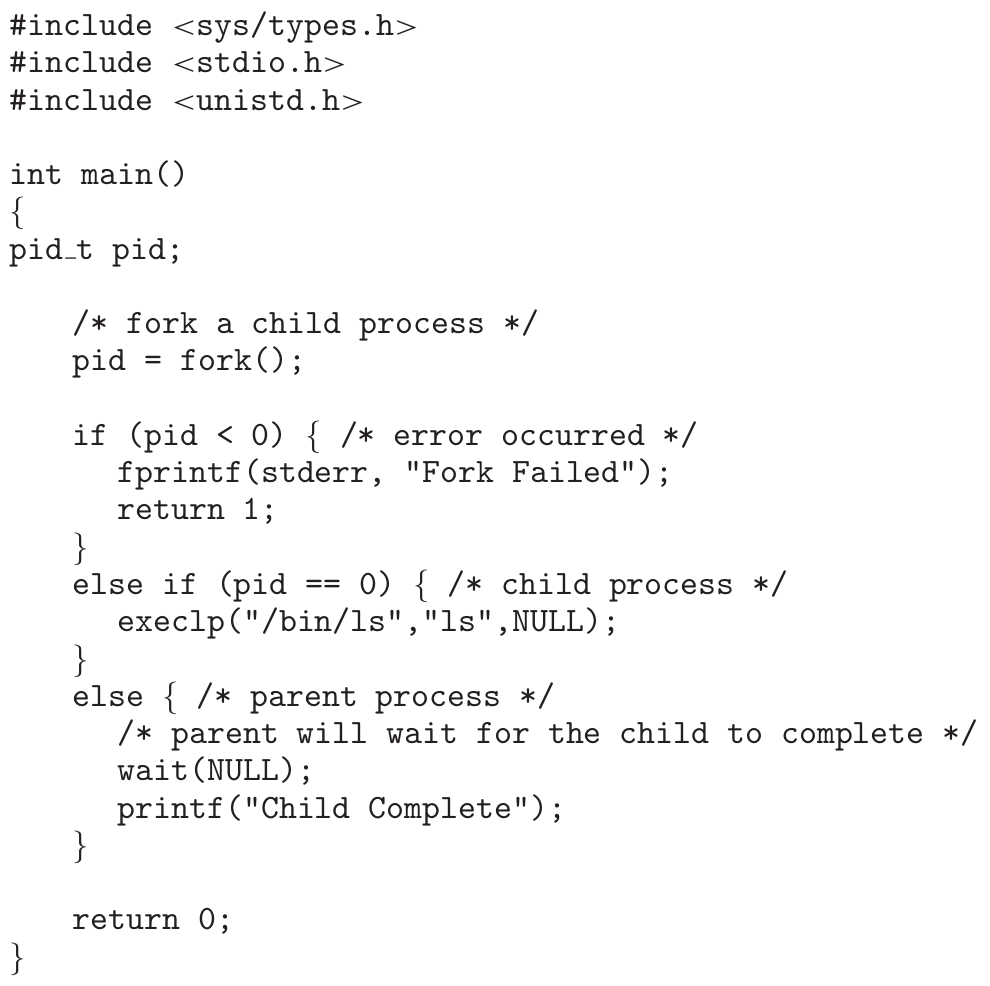
**Ćwiczenia** do realizacji na laboratorium (obowiązkowe dla wszystkich specjalności):

<https://docs.google.com/document/d/1DAMDgfjTGlftysS2aZYKFCVZbZtrvtCKZiEh_2FMLb0>

**Zadanie dla Cyberbezpieczeństwa**

**Zadanie 1. (tego nie wysyłamy)**

Proszę zapoznać się z działaniem programu (przekleić, skompilować, uruchomić):



Źródło: A. Silberschatz, *Operating Systems Concepts Essentials*

Podany program proszę rozbudować w następujący sposób:

* Po wykonaniu funkcji *fork()* odczytać i wyświetlić PID: zarówno procesu macierzystego, jak i potomnego (obie te informacje wyświetlić zarówno z poziomu procesu macierzystego, jak i potomnego).
* Zawiesić działanie programu macierzystego na funkcji odczytującej z klawiatury.
* Dla procesu potomnego zamienić funkcję *execlp()* na funkcję odczytania z klawiatury zmiennej całkowitej.
* Przed wykonaniem *fork()* zarezerwować zasoby:
  + Otworzyć do pisania plik
  + Zarezerwować miejsce w pamięci (statycznie i dynamicznie)
  + Uwaga (nowe): powyzsze wykonac zarowno dla zmiennych lokalnych, jak i globalnych.

Na tej podstawie proszę przedstawić:

* Odszukać w *pstree* procesy (macierzysty i potomny) i przedstawić je
* Zinterpretować dostęp do zasobów w procesie macierzystym i potomnym
* Który z procesów odczyta wprowadzone dane z klawiatury (dlaczego?) ?
* Co się wydarzy z procesem potomnym po usunięciu procesu macierzystego ?
  + W dwojakim postępowaniu:
  + 1/ proces macierzysty zakończył działanie
  + 2/ proces macierzysty otrzymał kill -9

**Zadanie na podsumowanie**

Proszę sprawdzić, ile procesów potomnych uda się utworzyć, zanim system ulegnie zawieszeniu ?

Proszę przedstawić działanie polecenia ulimit na przykładzie tego programu.

