



Laboratorium Informatyka 6

Opracował: mgr inż. Woźniak Piotr

1. Przetestuj poniższe polecenia związane z procesami w systemie Linux:
 - ps
 - top
 - pstree
2. Napisz program z pętlą nieskończoną, zaobserwuj jego działanie z pomocą menażera zadań i poprzednich poleceń. Zatrzymaj proces z użyciem komendy kill.
3. Przetestuj działanie poniższego kodu.

```
#include <unistd.h>
#include <stdio.h>
int main()
{
    pid_t pid;
    pid = fork();
    if(pid==-1)
    {
        printf("Błąd\n");
    }
    if(pid==0)
    {
        //kod procesu dziecka
        printf("Proces-dziecko\n");
    }
    else
    {
        //kod procesu rodzica
        printf("Proces-rodzic\n");
    }
    return 0;
}
```

4. Zmodyfikuj kod z zadania 3, aby zaobserwować działanie procesu dziecka i rodzica z użyciem menażera zadań.
5. Opracuj program tworzący 10 procesów, w którym każdy wyświetla własny identyfikator PID.
6. Przetestuj aplikację „two_processes_pipe.c”. Zwróć uwagę na jej działanie po zmodyfikowaniu czasu wysyłania informacji. Używając polecenia kill na jednym z procesów sprawdź działanie programu.

7. Stwórz program tworzący 3 procesy. Pierwszy wypisuje co jedną sekundę „generuje”, drugi generujący i wyświetlający liczę parzystą co 3 sekundy, a trzeci generujący i wypisujący liczbę nieparzystą co 6 sekund.
8. Przetestuj aplikację „two_process_sync.c”. i na jej podstawie zaproponuj program, który tworzy 3 procesy. Z użyciem semaforów pierwszy proces ma po wyświetleniu liczb od 1 do 10 odblokować 2 proces. 2 Proces po wypisaniu tekstu „2 proces” ma odblokować 3 proces, który ma wypisywać w nieskończoność „koniec”.

Kody aplikacji :

http://ssamolej.kia.prz.edu.pl/dydaktyka/inf_1EE_ZI/Procesy_wspolbieznosc1.zip