

Zadanie 1

Napisz program demonstrujący różne reakcje na przykładowy sygnał SIGUSR1 w zależności od ustawionych dyspozycji. Reakcja na sygnał SIGUSR1 programu powinna zależeć od wartości argumentu z linii poleceń. Argument ten może przyjmować wartości: none, ignore, handler, mask. Program w zależności od parametru odpowiednio: nie zmienia reakcji na sygnał, ustawia ignorowanie, instaluje handler obsługujący sygnał (działający w ten sposób, że wypisuje komunikat o jego otrzymaniu), maskuje ten sygnał oraz sprawdza czy wiszący/oczekujący sygnał jest widoczny. Następnie przy pomocy funkcji raise wysyła sygnał do samego siebie oraz wykonuje odpowiednie dla danej opcji, opisane wyżej działania.

Zadanie 2

Napisz dwa programy: sender program wysyłający sygnały SIGUSR1 i catcher - program odbierający te sygnały. Program catcher jest uruchamiany jako pierwszy, wypisuje swój numer PID i czeka na sygnały SIGUSR1. Po każdorazowym odebraniu sygnału SIGUSR1 przez program catcher powinno nastąpić potwierdzenie odbioru tego sygnału. W tym celu, catcher wysyła do sendera sygnał SIGUSR1 informujący o odbiorze sygnału. Sender powinien wysłać kolejny sygnał dopiero po uzyskaniu tego potwierdzenia. Czekanie na takie potwierdzenie może odbywać się wywołując funkcję sigsuspend. Wraz z każdym sygnałem przesłanym przez sender powinien zostać przesłany tryb pracy programu catcher za pomocą funkcji sigqueue. Możliwe tryby pracy:

1- wypisanie na standardowym wyjściu liczb od 1 do 100

2 - wypisanie na standardowym wyjściu liczby otrzymanych żądań zmiany trybu pracy od początku działania programu

3 – zakończenie działania programu catcher.

PID sendera catcher pobiera ze struktury siginfo_t po przechwyceniu od niego sygnału. Program sender jako pierwszy parametr przyjmuje PID procesu catcher. Drugi parametr określa tryb pracy procesu catcher - w jednym wywołaniu może być przekazany jeden taki tryb. Program catcher działa aż do momentu otrzymania trybu 3 (zakończenie działania programu catcher). Program sender działa do momentu otrzymania potwierdzenia otrzymania przez catcher przesłanego trybu, po czym kończy pracę. Program sender można wywołać wielokrotnie aż do zakończenia pracy przez catcher.

UWAGA! W żaden sposób nie opóźniamy wysyłania sygnałów, wszelkie "gubienie" sygnałów jest zjawiskiem naturalnym.