

## Sistemi operativi – laboratorio

### Esercitazione 4: segnali

12-13 Gennaio 2023

#### Ese\_1:

Scrivere un programma che stampa ininterrottamente a video ogni 5 secondi la stringa "Ciao, io sono un processo in esecuzione (PID)" (PID e' il process identifier del processo in esecuzione). Il programma deve bloccare tutti segnali di sistema, tranne il segnale SIGINT. Ogni qualvolta SIGINT viene inviato al processo in esecuzione, quest'ultimo stampa a video la stringa "Ctrl+C non mi puo' fermare!"

P.S.

Per terminare il processo in esecuzione, aprire un nuovo terminale ed inviare il segnale SIGKILL al processo con il comando *kill -SIGKILL PID* (PID e' il process identifier del processo in esecuzione).

#### Ese\_2:

Scrivere un programma che implementa il comando *sleep SECONDS* (vedi man sleep) utilizzando le system call *alarm(...)*, e *pause()*. Allo scadere di SECONDS secondi, il programma stampa a video la stringa: "Che bella dormita!"

#### Ese\_3:

Scrivere un programma che genera due processi figlio. Il processo figlio1 invia il segnale SIGUSR1 al processo parent. Il processo parent inoltra il segnale SIGUSR1 al processo figlio2. Il processo figlio2, dopo avere ricevuto SIGUSR1, stampa a video "<figlio2> SIGUSR1 ricevuto!", invia il segnale SIGUSR2 al processo parent, ed infine termina. Il processo parent inoltra il segnale SIGUSR2 al processo figlio1. Il processo figlio1, dopo avere ricevuto SIGUSR2, stampa a video "<figlio1> SIGUSR2 ricevuto!", ed infine termina. Il processo parent attende la terminazione di figlio1 e figlio2, ed infine termina.

#### Ese\_4:

Scrivere un programma che genera due processi figlio. Il processo figlio1 stampa ininterrottamente a video ogni 3 secondi la stringa "Sono figlio1, sto giocando..." (lavoro). Il processo figlio2 invia il segnale SIGSTOP a figlio1 ininterrottamente ogni 10 secondi (azione di disturbo). Il processo parent, monitora ininterrottamente i processi figli creati ogni 15 secondi (azione di monitoraggio). Se figlio1 e' nello stato STOP, il processo parent invia il segnale SIGCONT al processo figlio1 (azione di ripristino). Se l'utente invia il segnale SIGINT al processo parent, il processo parent termina i due processi figli inviando il segnale SIGTERM, ed infine termina.