

Sistemi Operativi – Lab 8 – 02.12.19 - A.A. 2019/2020 - Prof. L. Sterpone

Goal: sincronizzazione dei processi tramite segnali, uso della pipe.

Esercizio 1 (signal):

Scrivere un programma in linguaggio C in cui il processo padre (master) crea due processi figli (figlio 1 e figlio 2), attende la terminazione di entrambi, visualizza un messaggio e termina l'esecuzione.

Il processo figlio 1, apre un file di testo test_1.txt, legge il file una riga per volta e la stampa a video; il processo figlio 2, apre un file di testo test_2.txt, legge il file una riga per volta e la stampa a video.

Nota: Provare a eseguire tale esercizio in una seconda alternativa: supponendo di ottenere un output su schermo interlacciato dei due file.

Esercizio 2 (signal e kill):

Scrivere un programma C che attende all'infinito. Il processo padre dovrà gestire tramite apposito handler due segnali: SIGCHLD e SIGINT e comporarsi nel seguente modo:

- dovrà creare un processo figlio, attendere la ricezione del segnale SIGCHLD e terminare.
- attende anche la ricezione del segnale SIGINT (generato dall'esterno) e invia al figlio a sua volta lo stesso segnale

Il processo figlio si attiva alla ricezione del segnale SIGINT, stampa il suo PID e termina.

Esercizio 3 (pipe):

Scrivere un programma C in cui il padre crea un processo figlio, riceve dal figlio una sequenza di stringhe che visualizza e termina quando il figlio ha terminato la sequenza.

Il figlio dovrà aprire un file di testo, leggere il file una riga per volta, inviare la riga al padre e terminare quando non vi sono più caratteri da leggere nel file. Gestire la comunicazione tra padre e figlio attraverso una pipe.

Esercizio 4 (pipe e system call exec):

Scrivere due programmi in linguaggio C. Il programma prende una stringa in ingresso e la stampa in uscita utilizzando il comando *echo* attraverso una *exec*. Il secondo programma riceve una stringa dallo stdin e la stampa in uscita tutta maiuscola. Utilizzando obbligatoriamente ed esclusivamente questi due programmi convertire una stringa da minuscolo a maiuscolo.

Esercizio 5 (system call signal, kill, pause e sleep):

Si realizzi un programma C che crei due processi figli. Il figlio 1 invia il segnale SIGUSR1 al fratello passando per il padre. Il figlio 2 deve catturare il segnale inviatogli dal fratello e rispondere con il segnale SIGUSR2 sempre tramite il padre. Stampare a video delle stringhe di testo (e.g., come quelle utilizzate a lezione "Sono nel padre", "Sono nel figlio 1",...) per dimostrare il corretto funzionamento del programma.
