

Esercitazioni

29 Maggio 2008

Esercizio 1

Scrivere un Programma che:

scrive in memoria condivisa la stringa passata come primo parametro sulla linea di comando.

quindi genera un figlio e ne attende la terminazione, poi ne genera un altro di cui attende la terminazione, poi ne genera un altro ... e così via, fino a quando non riceve dall'ultimo figlio generato il segnale SIGUSR1, nel qual caso scrive il contenuto della memoria condivisa e termina.

Ogni figlio si comporta come segue:

se gli ultimi 4 caratteri della memoria condivisa contengono "ping" scrive di seguito "pong" altrimenti scrive di seguito "ping".

Se a questo punto la memoria condivisa contiene più di 30 caratteri, invia il segnale SIGUSR1 al padre e termina altrimenti termina

Esercizio 2

Scrivere un programma, prg, che:

- crea un segmento di memoria condivisa della dimensione di un intero e lo inizializza con il valore ottenuto come primo parametro sulla linea di comando. Crea una fifo di nome "FIFO".

- crea un semaforo e lo inizializza a 2 quindi setta un conto alla rovescia di 20 secondi, durante i quali esegue il seguente loop:

 - aspetta che il semaforo scenda a 0, quindi scrive il contenuto della memoria condivisa e poi riporta il semaforo al valore 2.

 - Quando i 20 secondi scadono prg legge dalla fifo due pid ed invia il segnale SIGINT ai processi corrispondenti, quindi termina.

Scrivere due altri programmi (prgA, prgB) che prendono su linea di comando gli id della shmem e del semaforo creati da prg, inviano sulla FIFO il proprio pid e poi si comportano come segue:

- prgA: esegue un loop infinito in cui decrementa di 2 il semaforo e poi quadruplica il contenuto della memoria condivisa.

- prgB: esegue un loop infinito in cui decrementa di 1 il semaforo e poi dimezza il contenuto della memoria condivisa.

Esercizio 3

Usando semafori e/o code di messaggi per la sincronizzazione, riscrivere il programma che copia un file usando due processi, uno che copia le vocali e uno che copia le consonanti.

Esercizio 4

```
void uscita(void)
{ printf("\n Ciao Ciao\n");}
void uscita2(void)
{printf("\n Good Bye\n");}
```

a) Qual e' l'output del programma?

```
int main(void)
{
    int pid = 0;
    atexit(uscita);

    pid = fork()

    if(pid == 0){
        atexit(uscita2);
        exit(0);}
    wait(NULL);
    kill(pid, USR1);
}
```

Esercizio 5

Scrivere un programma che crea un semaforo binario e lo setta a 0 e poi genera un figlio:

Il figlio funziona come segue:

aspetta che il semaforo diventi 1 e solo allora stampa a video "Posso agire"

Il padre aspetta che il figlio si fermi (stop) e solo allora pone il semaforo ad 1, quindi aspetta che il figlio riparta per porre nuovamente il semaforo a 0.

Eeguire il programma in background in modo da poter inviare il segnale di stop al figlio e vedere cosa succede.

Verificare se il figlio riesce a stampare la stringa "Posso agire"???