

## 1. primi passi

---

Si consideri la struttura seguente:

```
typedef struct my_data {  
    int counter;  
    char stringa[DATASZ];  
} data;
```

Scrivere un programma in cui il padre svolgerà le seguenti operazioni:

- creazione di un segmento di *SM* atto a contenere la struttura *data*;
- aggancio del segmento;
- scrittura all'interno del membro *stringa* di un messaggio qualsiasi e inizializzazione del membro *counter* a 0;
- sganciamento (*detach*) del segmento.
- dopo avere atteso la terminazione del figlio, il padre rimuoverà il segmento.

Il figlio svolgerà le seguenti operazioni:

- attacco del segmento di *SM* allo spazio di indirizzi del processo;
- stampa del contenuto dei due membri della struttura condivisa;
- sganciamento (*detach*) del segmento.

In particolare, riflettere su quando effettuare la *fork()*, e su soluzioni alternative per la creazione e il reperimento dell'identificatore del segmento di *SM*.

## 2. scrittore e lettore

---

Scrivere due programmi che interagiscano nella scrittura e lettura, come descritto a lezione. Lo scrittore deve leggere il contenuto del file *file\_prova* e scriverlo nella memoria condivisa; il lettore legge quanto scritto nella memoria condivisa e lo stampa a video.

I due programmi utilizzano due semafori per sincronizzarsi: uno per organizzare l'attività di scrittura e uno per la lettura.

Esempio di interazione con i due programmi:

```
$ ./writer < file_prova &  
[1] 87902  
$ ./reader  
// stampa a video il contenuto del file file_prova ...  
[lettore] : ricevuti 236 bytes (1 n_cycles)  
[scrittore]: inviati 236 bytes (1 n_cycles)
```

```
[1]+  Done      ./writer < file_prova
```

### 3. utilizzo della memoria condivisa

---

Scrivere un programma che riceve in input due numeri  $N$  ed  $M$ ; il programma alloca un'area di  $SM$  e poi crea  $N$  figli. Ciascun figlio cicla  $M$  volte e ad ogni ciclo:

- attende un tempo random tra 0 e 9 secondi;
- accede in mutua esclusione alla  $SM$  e scrive dentro il suo  $PID$ .

Il padre cicla  $M$  volte e ad ogni ciclo:

- attende 5 secondi;
- stampa su terminale il  $PID$  che trova scritto nella  $SM$ .

Quando un figlio ha finito gli  $M$  cicli termina. Quando il padre ha finito gli  $M$  cicli si mette in attesa della terminazione dei figli, rimuove l'area di  $SM$  e termina.