

Progetto di sistemi operativi: Reazione a catena

Francesco Mauro, Riccardo Oro

A.A 2023/2024

Contents

1	Processo master	1
2	Processo atom	2
3	Processo psu	2
4	Processo activator	2
5	Processo inhibitor	2

1 Processo master

- Si occupa di leggere da file attraverso la funzione `scan_data()` i valori di configurazione necessari per la simulazione che verranno associati alla struct `config`.
- Crea `NATOMIINIT` atomius andola funzione `fork`. Il processo figlio generato avrà associato un array con
- Crea il processo *inhibitor* sotto richiesta dell'utente attraverso la funzione `fork`. Il processo figlio generato avrà associato un array contenente i parametri di configurazione personalizzati e il PID del processo master, eseguirà un `execvp` a `./bin/inhibitor`.
- Si occupa di registrare i segnali che verranno gestiti dagli appositi handler, che permettono la gestione delle terminazioni della simulazione.
- Dopo aver inizializzato tutto il necessario, imposta un alarm con valore `SIM_DURATION` che rappresenta la durata complessiva della simulazione. Il segnale `SIGALARM` gestito
- Ogni secondo mostra a schermo:
 - Il numero di fissioni avvenute nell'ultimo secondo.
 - Il numero di scorie raccolte nell'ultimo secondo.
 - L'energia prodotta dei processi atomo nell'ultimo secondo.

- I bilanciamenti effettuati di processo inhibitor per limitare le fissioni dei processi atomo nell'ultimo secondo.
- Si occupa, in seguito allo scadere di un timer visibile a schermo, di far partire i processi atomi per la simulazione.
- Si occupa di gestire le terminazioni della simulazione.

2 Processo atom

Il processo atomo viene generato attraverso fork dal processo master. Dopo la sua nascita fa le seguenti operazioni:

- Inizializza gli *IPC* necessari.
- Registra il segnale *SIGCHLD* associandolo all'handler apposito.
- Registra il segnale *SIGUSR1* associandolo ad un handler che si occuperà, in caso di *MELTDOWN*, di inviare il segnale *SIGUSR1* al processo master che gestirà la terminazione della simulazione.
- Preleva e associa all'apposita struct i valori della configurazione attraverso *argv*.

Dopo queste operazioni il processo atomo si autoinvia un segnale *SIGSTOP* per indicare che ha inizializzato tutto il necessario, attendendo dal processo master il segnale *SIGCONT* che arriverà quando scadrà il timer a schermo e verrà impostato l'alarm con *SIMDURATION*. Successivamente, il processo atomo si mette all'opera, dopo aver ricevuto il segnale *SIGCONT*. Nel caso in cui il suo numero atomico sarà inferiore a quello prestabilito all'interno della configurazione attraverso il parametro *MIN_ATOMICO*, esso non potrà effettuare fissione e

3 Processo psu

TODO

4 Processo activator

TODO

5 Processo inhibitor

TODO