

Sistemi operativi – laboratorio

Esercitazione 5: shared memory

03-06 Aprile 2023

Ese_1:

Dati due array di pari dimensione di definisca l'operazione $ps(v1, v2)$ come:

$$v1[0] * v2[0] + v1[1] * v2[1] + \dots v1[n] * v2[n]$$

Realizzare un'applicazione C basata su memoria condivisa e più processi (1 padre e tanti figli del processo padre) per parallelizzare il calcolo sopra descritto.

Ogni processo figlio dovrà svolgere il conteggio $v1[x] * v2[x]$ a lui assegnato, mentre il processo padre svolgerà la somma finale e stamperà il risultato complessivo.

Ese_2:

Realizzare un'applicazione Client-Server basata su memoria condivisa per leggere i primi 100 caratteri di un file presente sul filesystem. Il processo Server istanzia un segmento di memoria condivisa SH1 grande a sufficienza per contenere la seguente struttura:

```
struct Request {  
    char [250]; // path di un file nel filesystem  
    key_t shmkey; // shmkey di un segmento di memoria condivisa SH2  
}
```

Appena un'istanza di Request e' depositata in SH1, il programma Server esegue le seguenti operazioni:

- a) apre in lettura il file indicato in pathname;
- b.1) deposita in SH2 i primi 100 caratteri del file aperto;
- b.2) se il file non esiste, deposita in SH2 il valore -1;
- c) rimuove il segmento SH1
- d) infine, termina

Il processo Client crea un segmento di memoria condivisa SH2 grande 100 bytes. Chiede all'utente il pathname di un file, e deposita in SH1 la struttura Request. Appena i primi 100 bytes del file richiesto sono depositati in SH2, il programma Client esegue le seguenti operazioni:

- a) stampa a video il contenuto di SH2
- b) rimuove il segmento SH2
- c) rimuove i semafori utilizzati per la sincronizzazione con il Server
- d) termina

Ese_3:

Estendere l'esercizio Ese_1 in modo tale che il processo Client possa leggere l'intero file utilizzando il segmento di memoria condivisa SH2.