

Sistemi Operativi CANALE M-Z

Compito Scritto del 1 Marzo 2022

Cognome=_____, Nome=_____
Matricola=_____

Durata 3 ore

Domande 1: Esercizi al Calcolatore

Domanda 1.a: max 4 punti

Scrivere un programma che crea due processi (uno padre e uno figlio) ed utilizzi i parametri del main() argc e argv. Entrambi i processi scrivono su uno stesso file condiviso, il cui nome viene passato come primo parametro. Il padre scrive "Come ". Il processo figlio scrive "stai?". Alla fine del programma il file di testo deve contenere la frase "Come stai?".

L'esercizio deve essere risolto senza l'uso dei semafori.

Domanda 1.b: max 4 punti

Scrivere un programma che crea cinque processi (uno padre e quattro figli). Il processo padre, dopo aver generato i processi figli, invia il segnale SIGUSR1 a tutti i processi figli. Si supponga che dopo aver inviato tale segnale ai processi figli, il processo padre esegua un ciclo for con un numero elevato di iterazioni (il numero può essere liberamente scelto dallo studente, facendo in modo che il tempo di esecuzione del ciclo for sia di qualche decina di secondi).

Ogni processo figlio, attende l'arrivo del segnale SIGUSR1. All'arrivo del segnale SIGUSR1, stampa il proprio PID e termina.

È richiesta la gestione della terminazione dei processi figli tramite l'uso dei segnali, definendo una opportuna funzione di SignalHandler che eviti che i figli diventino zombie.

Domanda 1.c: max 4 punti

Scrivere un programma che inizialmente alloca un vettore dinamico di interi (di lunghezza passata tramite i parametri argc e argv[] del main). A questo punto il programma (main thread) si occupa di riempire in modo casuale il vettore dinamico. Una volta riempito il vettore, il main lancia tre threads ciascuno dei quali eseguono la somma dei valori contenuti nelle 3 porzioni di vettore. Si supponga di dividere idealmente il vettore in 3 parti uguali; ciascun thread si occuperà di una di queste 3 porzioni. Immaginando una gestione sincrona, quando i 3 threads finiscono, il main thread calcola la somma dei 3 valori ottenuti ed effettua la stampa a video e termina.

Domanda 1.b: max 6 punti

Sviluppare un'applicazione client/server, realizzando due programmi diversi (uno client e l'altro server) caratterizzati dal seguente scambio di dati. Il processo client invia un array di 10 caratteri inseriti da tastiera. Il server fornisce al client un valore intero corrispondente al numero di caratteri 'A' presenti nel vettore.

Lo scambio dati tra ogni client e il server continua fino a quando il client non si disconnette. Verrà considerato errore se il numero di scambi dati tra client e dal server è pari ad 1. Si supponga di utilizzare i thread nel Server per la gestione di più client.

Domande 2: Teoria

Domanda 2.a: max 4 punti

Descrivere in modo dettagliato i tre differenti modelli per la gestione dei thread user-level:

- Modello multi-a-uno (M:1)
- Modello uno-a-uno (1:1)
- Modello multi-a-molti (M:M)

Risposta: Si crei un file di testo nel computer, denominato “Domanda2a.txt”, che dovrà essere consegnato insieme ai programmi al punto precedente.

Domanda 2.b: max 4 punti

Descrivere in modo approfondito il problema della sincronizzazione noto come “**Produttore/Consumatore**”.
Descrivere le operazioni svolte dai processi produttore e consumatore.

Risposta: Si crei un file di testo nel computer, denominato “Domanda2b.txt”, che dovrà essere consegnato insieme ai programmi al punto precedente.

Domanda 2.c: max 4 punti

Descrivere in modo dettagliato l’algoritmo di schedulazione CFS di Linux.

Risposta: Si crei un file di testo nel computer, denominato “Domanda2c.txt”, che dovrà essere consegnato insieme ai programmi al punto precedente.