Sistemi Operativi

Compito del 20-06-2023

Esercizio 1 (10 punti)

In un sistema sono presenti quattro processi P_1, P_2, P_3 e P_4 che usano quattro tipi di risorsa A, B, C, D di cui sono presenti 2 unità per ciascun tipo. Ogni processo usa al massimo una istanza di ogni tipo di risorsa e hanno la seguente matrice di allocazione delle risorse:

	Allocazione			
	A	**B**	**C**	**D**
P_1	0	1	1	1
P_2	1	1	0	1
P_3	1	0	0	0
P_4	0	0	0	0

Usando l'**algoritmo del banchiere** stabilire se il sistema si trova in uno stato sicuro ed in caso positivo indicare una sequenza sicura. Quindi indicare per ogni processo se in questo stato le possibili richieste di risorse disponibili portano il sistema in uno stato sicuro o meno (giustificare il risultato).

Esercizio 2 (20 punti)

Si vuole realizzare il seguente sistema:

sono presenti N thread Generator che generano ognuno un valore e inseriscono in una coda un messaggio con id del generatore ed il valore quindi attendono per X millisecondi. La coda è limitata ad L messaggi.

Sono presenti M thread Worker dove ognuno iterativamente preleva atomicamente N messaggi dalla coda li elabora in un tempo variabile in [T,T+D) e inserisce il risultato nel OutputManager aspettando se il risultato precedente non è stato acquisito.

Sono presenti due OutputThread dove ognuno preleva dal OutputManager gli M risultati prodotti dagli Worker e li stampano.

Per facilitare il testing i Generator generano tutti la sequenza dei numeri interi partendo da 1 e gli worker calcolano la somma dei valori presenti negli N messaggi.

Il programma principale deve far partire i thread necessari e dopo 10 secondi terminare tutti i thread e stampare: per ogni Generator il numero di messaggi generati ed il totale dei messaggi generati da tutti i Generator, per ogni Worker il numero di elaborazioni fatte e per ogni OutputThread il numero di stampe fatte.

Realizzare in java il sistema descritto usando i ***semafori*** per la sincronizzazione tra thread (sarà considerato errore usare polling o attesa attiva).

Sistemi Operativi

Compito del 20-06-2023

Exercise 1 (10 points)

In a system there are four processes P_1, P_2, P_3 and P_4 which use four resource types A, B, C, D of which there are two units of each type. Each process uses at most one instance of each resource type and they have the following resource allocation matrix:

	Allocation			
	A	**B**	**C**	**D**
P_1	0	1	1	1
P_2	1	1	0	1
P_3	1	0	0	0
P_4	0	0	0	0

Using the **banker's algorithm**, determine if the system is in a secure state and if so, indicate a secure sequence. Then indicate for each process whether in this state the possible requests for available resources lead the system to a safe state or not (justify the result).

Exercise 2 (20 points)

We want to create the following system:

there are N Generator threads that each generate a value and queue a message with the generator id and the value, then wait for X milliseconds. The queue is limited to L messages.

There are M Worker threads where each one iteratively fetches N messages atomically from the queue, processes them in a variable time in [T,T+D) and inserts the result in the OutputManager waiting if the previous result has not been acquired.

There are two OutputThreads where each takes from the OutputManager the M results produced by the Workers and print them.

To facilitate testing, the Generators generate the sequence of integers starting from one and the Workers calculate the sum of the values present in the N messages.

The main program must start the necessary threads and after 10 seconds terminate all the threads and print: for each Generator, the number of messages generated and the total of messages generated by all the Generators, for each Worker the number of processings done and for each OutputThread the number of prints made.

Implement the described system in java using **semaphores** for synchronization between threads (it will be considered an error to use polling or active waiting).