Sistemi Operativi

Compito 22-11-2021

Esercizio 1 (10 punti)

Un sistema operativo adotta la politica di scheduling dei thread a code multiple con *prelazione* tra le code. Sono presenti due code una a bassa priorità con scheduling round robin con quanto q=3 ms ed una ad alta priorità con scheduling round robin con quanto q=2ms. Inoltre se un thread a bassa priorità ha un CPU burst inferiore al quanto viene promosso nella coda ad alta priorità. Se un thread viene prelazionato viene rimesso in testa alla coda e quando riprenderà l'esecuzione riceverà la parte restante del quanto che non ha utilizzato.

Il sistema deve schedulare i seguenti thread con tempi di arrivo, priorità e uso CPU/IO indicati:

T_1 T_{arrivo}=0 pri=L CPU(2ms)/IO(6ms)/CPU(3ms)
T_2 T_{arrivo}=1 pri=L CPU(4ms)/IO(3ms)/CPU(2ms)
T_3 T_{arrivo}=2 pri=L CPU(1ms)/IO(3ms)/CPU(3ms)
T_4 T_{arrivo}=3 pri=L CPU(4ms)/IO(3ms)/CPU(2ms)

Si determini: il **diagramma di Gantt**, il **tempo di attesa** medio, il **tempo di ritorno** medio, il **tempo di risposta** medio e il numero di cambi di contesto. Inoltre indicare in ogni istante il numero di processi in attesa nelle code dei processi pronti.

Esercizio 2 (20 punti)

In un ambiente sono presenti N persone che si muovono sul piano x/y, ogni persona è caratterizzata da una posizione x,y (double), una carica virale >=0 e se è o meno vaccinato. Se la carica virale è > 10 contagia tutte le persone vicine entro 2 metri incrementando di 5 la carica virale del vicino se questo non è vaccinato e con una probabilità del 10% se vaccinato. Se la carica virale supera 100 la persona si ferma e va in ospedale che però ha la capacità massima M. La persona che va in ospedale esce dall'ambiente condiviso e richiede di entrare in ospedale e vi permane per un tempo fisso T, quindi azzera la propria carica virale e ritorna nell'ambiente condiviso. Ogni persona iterativamente prima contagia i vicini, quindi aggiorna la propria posizione di un valore dx in [-1,1) e dy in [-1,1), poi diminuisce di 1 la carica virale (se > 0) e poi aspetta 100ms. Si noti che una persona può contagiare la stessa persona in momenti successivi e che una persona può essere contagiata contemporaneamente da più persone.

Nel programma principale posizionare tutte le N persone in una posizione casuale in [0, 20)x[0,20), impostare la carica virale iniziale a 50 per K persone (K<N) e a zero per le restanti e impostare come vaccinate altre V persone (tra quelle con carica virale a 0) (V>=0, V<=N-K).

Far partire i thread necessari e quindi ogni secondo stampare il numero di persone con carica virale > 10, il numero di persone in ospedale ed il numero di persone in attesa di entrare in ospedale.

Dopo 30 secondi terminare tutti i thread e stampare per ogni persona il numero di contagi fatti e ricevuti ed il numero di volte stato in ospedale.

Realizzare in java il sistema descritto usando i ***semafori*** per la sincronizzazione tra thread.