Sapienza - Università di Roma
Facoltà di Ingegneria dell'Informazione, Informatica e Statistica
Corsi di Laurea in Ingegneria Informatica ed Automatica ed Ingegneria dei Sistemi Informatici
Esercitazioni di Progettazione del Software - A.A. 2010/11

Esercitazione sulla programmazione concorrente e sull'uso dei thread in Java

Esercizio 1. Realizzare in Java un'applicazione che simuli una gara di corsa tra un numero n (fornito in input) di corridori, ciascuno caratterizzato dal proprio nome. I corridori sono modellati da thread Java. Ognuno di essi, dopo l'avvio, incrementa iterativamente un contatore, stampando su schermo, ad ogni iterazione, il proprio nome ed il numero di iterazioni eseguite (cio i metri percorsi). Dopo aver eseguito 100 iterazioni, il thread stampa su schermo un messaggio di notifica del proprio arrivo.

1. Dopo aver eseguito il programma Java, analizzarne l'output e stabilire se l'esecuzione dei vari thread sia avvenuta in maniera sequenziale (ovvero viene completato il primo thread avviato, poi il secondo, e così via) o meno;
2. Se dall'output emerge un comportamento sequenziale, individuarne le cause e proporre una modifica al programma che le rimuova;
3. Dopo aver completato i punti precedenti, analizzare la porzione di codice in cui i thread vengono avviati e verificare se si possa verificare la condizione in cui *alcuni thread sono all'interno del ciclo di conteggio mentre altri devono essere ancora avviati (o istanziati)*. In caso di risposta affermativa, proporre una modifica al programma che escluda tale eventualità.

Suggerimenti: utilizzare un `Bang` che avvia tutti i thread solo dopo che tutti quanti sono stati istanziati/avviati, ed un `ContatoreSincronizzato` per tenere il conteggio di quanti thread mano mano vengono istanziati/avviati. Attenzione a sincronizzare tale contatore, dato che è condiviso tra tutti i thread.

Esercizio 2. Si consideri l'esercizio precedente. Di esso si vuole creare una versione modificata, in cui, oltre ai corridori, è presente un giudice, anch'esso rappresentato da un thread. La gara si svolge come segue:

- i corridori vengono avviati (tenere presenti le osservazioni dell'esercizio precedente);
- il giudice viene avviato, dopodichè rimane in attesa che il primo corridore raggiunga il traguardo;
- il primo corridore che raggiunge il traguardo memorizza un riferimento all'oggetto che lo rappresenta in un'opportuna struttura ordinata (che rappresenterà la classifica);
- quando il primo corridore giunge al traguardo, il giudice ne annuncia la vittoria stampando un messaggio su schermo, quindi termina;
- ogni altro corridore, quando arriva al traguardo, memorizza il proprio arrivo, accodando un riferimento all'oggetto che lo rappresenta nella struttura ordinata;
- dopo che *tutti* i corridori hanno terminato la propria esecuzione, la classifica di arrivo viene stampata su schermo.

Suggerimenti: utilizzare una `Corsa` (che tra le sue proprietà mantiene la classifica di arrivo) ed un `Giudice`. Fare attenzione all'uso del costrutto di `join` tra thread al fine di tempificare opportunatamente i thread dei corridori e del giudice.