

L'obiettivo di questa serie d'esercizi è d'imparare ad utilizzare le tecniche e gli strumenti che permettono d'eseguire un'applicazione in parallelo, sfruttando i lightweight-tasks e l'executor framework di Java.

Esercizio 1

Copiate il codice sorgente 'S10Esercizio1.java' e analizzatelo nel dettaglio. In seguito, provate a modificare il programma in modo che utilizzi l'executor framework (tramite thread-pools pre-configurati) per eseguire le operazioni di moltiplica delle matrici in parallelo.

Esercizio 2

Riprendete il codice sorgente dell'esercizio 1 ed estendetelo in modo che i tasks, dopo ogni moltiplica delle matrici, calcolino la somma di tutti gli elementi contenuti nella matrice risultante dalla moltiplica. Il programma dovrà recuperare tutte le somme di tutte le moltipliche e, prima di terminare, dovrà stampare a schermo il valore più grande fra tutte le somme. Per fare in modo che i tasks responsabili delle moltipliche restituiscano i valori delle somme, interrogate le Futures in modo diretto tramite il metodo `get()`.

Esercizio 3

Riprendete il codice sorgente dell'esercizio 2 e rifattorizzatelo in modo che il programma, al posto di usare le Futures in modo diretto, sfrutti il `CompletionService`.

Esercizio 4

Copiate il codice sorgente 'S10Mandelbrot.java' e analizzatelo nel dettaglio. In seguito provate a modificare il programma, in modo che utilizzi l'executor framework di Java.

Esercizio 5 - Performance

Riprendete il codice dell'esercizio 4 e modificate il codice introducendo le misure dei tempi necessarie per analizzare le performances dell'applicazione. In seguito, provate ad aumentare il numero di threads utilizzati dall'applicazione e calcolate lo speed-up ottenuto. Realizzate un grafico di scalabilità dell'applicazione. Analizzate e descrivete i risultati ottenuti.