

Sistemi Operativi e Reti – Estratto dalla prova di Sistemi Operativi del 25 Marzo 2009 – Tempo a disposizione 2.5 ore

Aritmetica Vettoriale

Si vuole scrivere una libreria che implementi una serie di operazioni tra vettori, da eseguire in parallelo. In particolare la libreria deve consentire di eseguire la somma e la sottrazione tra due vettori (di pari lunghezza), facendo uso dei processori a disposizione. A tal proposito si può usare la funzione predefinita **Runtime.getRuntime().availableProcessors()** che restituisce il numero di processori a disposizione sul PC in cui gira il programma da progettare.

Si progetti quindi la classe **Operazioni** preposta allo scopo.

La classe deve essere dotata del metodo **int[] somma(int[] V1, int[] V2)** che ritorna un nuovo vettore dato dalla somma elemento per elemento dei vettori puntati V1 e V2, e dell'analogo metodo **int[] sottrai(int[] V1, int[] V2)**.

Ad esempio se $A = [1,4,7,8,12]$ e $B = [1,2,1,2,1]$, allora $\text{somma}(A,B) = [2,6,8,10,13]$, mentre $\text{sottrai}(A,B) = [0,2,6,6,11]$.

La classe deve essere progettata in maniera tale da far uso del numero di thread ideale: ad esempio se **getProcessori() = 4**, può convenire effettuare la somma in parallelo usando 4 thread.

Bonus: si progettino **somma** e **sottrai** in maniera tale da poter essere invocate in parallelo non solo dal programma main, ma da eventuali altri processi thread. Ad esempio se in un certo momento è in corso l'operazione **somma(A,B)**, deve essere possibile che un thread invochi ulteriormente **somma** o **sottrai** senza interferire con l'elaborazione in corso. La classe **Operazioni** deve comunque garantire che il numero di thread in esecuzione in un certo momento non superi il numero di processori disponibili.

E' parte integrante dell'esercizio completare le specifiche date nei punti non esplicitamente definiti, e risolvere eventuali ambiguità.