

Laborator 1 (PPD)

Implementati in Java

- A) adunarea a doua matrice de dimensiune (nxm)
 - B) inmultirea a doua matrice de dimensiune (nXk), respectiv (nXm)
- folosind multithreading.

-Numarul de threduri p trebuie sa fie un parametru care poate fi citit (modificat) inainte de inceperea executiei.

Datele de intrare corespunzatoare elementelor matricilor se vor citi din fisiere (care au fost anterior create folosind generare aleatoare de numere)!

Programul va afisa la sfarsit timpul global(total) T de executie corespunzator operatiei de adunare(respective operatiei de inmultire) de matrice.

($T = \text{timp_evaluat_dupa_terminarea_executiei_tuturor_threadurilor} - \text{timp_evaluat_inainte_de_pornirea_threadurilor}$)

Important: folositi o incarcare echilibrata de calcul pe fiecare thread.

Balanced Distribution (dimensiunea datelor pe care lucreaza fiecare thread este aproximativ egala).

- distributie pe linii
- distributii pe coloane
- distributie "perfecta" diferenta de nr de date corespunzatoare unui thread ≤ 1

Testarea va include:

-variante orientate pe testarea corectitudinii (dimensiuni relativ mici; de exemplu $< 10 \times 10$)

-variante orientate pe masurarea performantei (dimensiuni relativ mari; de exemplu $> 1000 \times 1000$);

Pentru fiecare testare includeti in documentatie informatii despre:

Dim_matrice | Nr_threaduri | Timp de executie | Obs- referitoare la sistemul pe care s-a facut testarea

Deadline:

Saptamana 3