**Documentație Laborator 1**

**Muntea Andrei – Marius (235)**

***I Cerință***

Implementați în Java, folosind multithreading:

1. Adunarea a două matrice de dimensiune (n x m)
2. Înmulțirea a două matrice de dimensiune (n x k), respectiv (k x m)

Numărul de threaduri *p* trebuie să fie un parametru care poate fi modificat înainte de începerea execuției. Datele de intrare se vor citi din fișiere care au fost anterior create folosind generare aleatoare de numere. Programul va afișa la sfârșit timpul global *T* de execuție corespunzător operației de adunare respectiv de înmulțire de matrice.

***II Descrierea soluției***

Vom implementa o clasă *Matrix* care va încapsula numărul de linii, numărul de coloane respectiv matricea de date. Pentru a folosi o distribuție cât mai balansată a datelor ca datele pe care operează fiecare thread să fie aproximativ egale, vom implementa operația de adunare respectiv de înmulțire între matrice specificând indicele liniei de început respectiv indicele liniei de sfârșit. Fiecare thread va procesa *nrLinii / nrThreaduri (+1)* linii.

Vom avea de asemenea o clasa generică *WorkerThread*<T> care moștenește clasa *Thread* și care suprascrie metoda *run()*. Vom trimite așadar un *ThreadContext* care va încapsula cele două matrice, indicele liniei de început, indicele liniei de sfârșit, precum și matricea în care se va salva rezultatul operației.

***III API-uri folosite***

public static void Multiply ( public static void Add (

\_In\_ Matrix firstMatrix, \_In\_ Matrix firstMatrix,

\_In\_ Matrix secondMatrix, \_In\_ Matrix secondMatrix,

\_Out\_ Matrix result, \_Out\_ Matrix result,

\_In\_ Integer startLine, \_In\_ Integer startLine,

\_In\_ Integer endLine \_In\_ Integer endLine

) )

***IV Rezultate testare***

|  |  |  |
| --- | --- | --- |
|  | ***Single Threaded*** | ***Multi Threaded*** |
| Sumă pe matrice de 3x3 cu date fixate | 0.03 (ms) | 0.23 (ms) - 2 threaduri |
| Sumă pe matrice de 7x7 | 0.41 (ms) | 2.30 (ms) - 5 threaduri |
| Sumă pe matrice de 1000x1000 | 38.63 (ms) | 312.42 (ms) - 7 threaduri |
| Înmulțire de matrice pătratice de 3x3 cu date fixate | 0.04 (ms) | 1.20 (ms) - 2 threaduri |
| Înmulțire de matrice 5x7 cu matrice de 7x2 | 0.03 (ms) | 0.45 (ms) - 3 threaduri |
| Înmulțire de matrice 782x852 cu matrice de 852x981 | 35207.19 (ms) | 8217.42 (ms)-11 threaduri |

