Laborator 4

*Documentație*

*Enunț*

Se considera n polinoame reprezentate prin lista de monoame.

Se cere adunarea polinoamelor folosind o implementare multithreading (p threaduri).

*Abordare*

Consideratii generale:

- reprezentarea unui polinom in memorie: lista inlantuita (1 nod=1monom) ordonata dupa exponentii monoamelor cu urmatorul INVARIANT (predicat adevarat la orice moment al executiei) de reprezentare:

-monoamele sunt ordonate dupa exponenti

-nu se pasteaza in lista monoame cu coeficient 0;

- nu exista doua noduri (monoame) cu acelasi exponent

- polinoamele se citesc din fisiere – cate un fisier pentru fiecare polinom;

- un fisier contine informatii de tip (coeficient, exponent) pentru fiecare monom al

unui polinom,

- fisierele input se creeaza prin generare de numere aleatoare.

Se porneste prin crearea unei liste inlantuita - L corespunzatoare unui polinom nul.

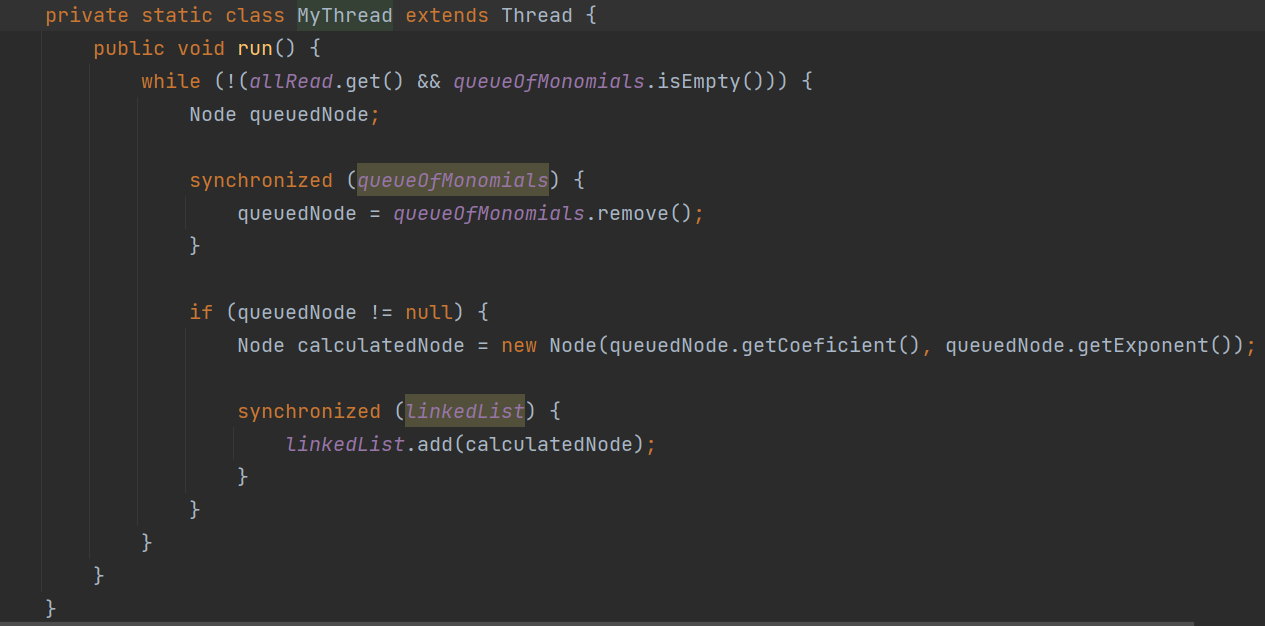
In final aceasta lista va continue polinomul rezultat.

Secvențial

Se citesc pe rând monoamele din fișiere, se creează un Node și se adaugă în lista înlănțuită linkedList, respectând invariantul.

Paralel

Primul proces citește toate monoamele, le adaugă într-o coadă queueOfMonomials, după care așteaptă celelealte N-1 threaduri (N = nr. total de threaduri alocate) să termine adunările. La final scrie în fișierul destinație polinomul obținut în linkedList.

Celelalte N-i threaduri, de tipul MyThread, așteaptă noi valori în coadă și, atât timp cât mai sunt monoame de citit, așteaptă noi valori. Apoi, citesc din coadă câte un monom și îl adaugă în linkedList, conform codului următor :

|  |  |  |
| --- | --- | --- |
| **Tip metodă** | **Nr procese** | **Timp executie** |
| secvențial | - | 30,60865 |
| Paralel | 4 | 132,9319 |
| 6 | 138,11647 |
| 8 | 133,42996 |

La final, funcția main verifică corectitudinea datelor comparând fișierul rezultat cu fișierul obținut în urma rulării secvențiale. Timpii rezultați sunt după cum urmează:

|  |  |  |
| --- | --- | --- |
| **Tip metodă** | **Nr procese** | **Timp executie** |
| secvențial | - | 39,19284 |
| Paralel | 4 | 127,82724 |
| 6 | 141,3617 |
| 8 | 142,02083 |

*Cazul 1 (10 polinoamen fiecare*

*cu gradul maxim 1000 si cu maxim 50 monoame)*

*Cazul 2 (5 polinoame, fiecare cu gr. max. 10000 si cu max. 100 monoame)*

*Observații*

Execuția secvențială este mai rapidă decât execuția paralelă.

Se obțin diferențe mici în cazul creșterii numărului de threaduri folosite.

Cazul 2, având polinoame cu un număr mai mare de monoame, înregistrează timpi defavorabili primului caz.