

Documentatie

Analiza

Implementarea C++ este de aproximativ 2 ori mai rapida decat cea Java.

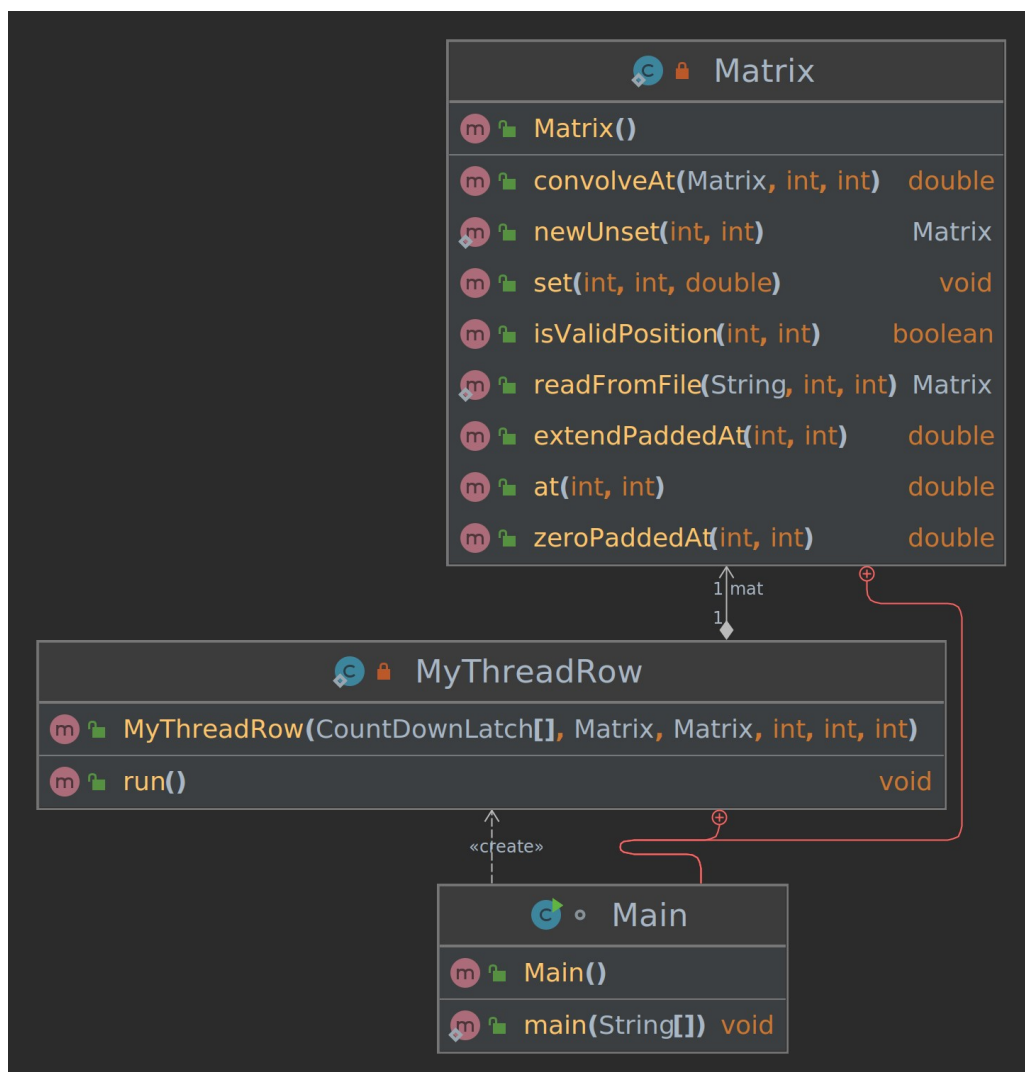
Fiecare thread alocă 3 vectori, fiecare dintre vectori având aceeași lungime cu lungimea unei linii în matrice. Asadar, complexitatea spațiu este $O(p \cdot 3 \cdot N)$; deoarece p și 3 sunt constante, complexitatea spațiu se încadrează în $O(N)$.

Implementare

Thread-urile parcurg ciclic liniile din matrice (din p în p). Fiecare linie din matrice este protejată cu un latch, care este inițializat cu numărul de thread-uri. Atunci când un anumit latch ajunge la 0, se consideră că linia pe care o protejează latch-ul poate fi suprascrisă.

Un thread aflat la linia i decrementează mai întâi toate latch-urile de la liniile care au indicii mai mici decât $i-1$. Apoi își copiază liniile $i-1$, i , și $i+1$, și așteaptă la latch-ul liniei i . Atunci când latch-ul liniei i se eliberează, calculează valoarea convoluției pe întreaga linie i , folosind liniile pe care și le-a salvat.

Diagrama UML



Tabel rulari

mat_size	p	runner	elapsed (seconds)
N=10,M=10	0	cpp	0.00
		java	0.00
	2	cpp	0.00
		java	0.00
N=1000,M=1000	0	cpp	0.02
		java	0.13
	2	cpp	0.03
		java	0.17
	4	cpp	0.03
		java	0.24
	8	cpp	0.03
		java	0.17
	16	cpp	0.04
		java	0.20
N=10000,M=10000	0	cpp	1.59
		java	2.98
	2	cpp	1.95
		java	4.19
	4	cpp	1.92
		java	4.27
	8	cpp	2.02
		java	4.43
	16	cpp	2.03
		java	4.38