Timpii de execuție:

* Varianta 0:

|  |  |
| --- | --- |
| Numar 1 = “123456789123456789” = Numar2 | 1.4513 ms |
| N\_1=1000 si N\_2=1000 (random digits) | 200.383 ms |
| N\_1=100 si N\_2=100000 (random digits) | 15549.8 ms |
| N\_1=N\_2=16 ; Numar1=”9999 4444 9999 9999” Numar1=”9999 5555 9999 9999” | 1.3004 ms |

* Varianta 1:

|  |
| --- |
|  |

|  |  |  |
| --- | --- | --- |
| Numar 1 = “123456789123456789” = Numar2 | 4  8  16 | 17.0264 ms  11.8062 ms  28.5505 ms |
| N\_1=1000 si N\_2=1000 (random digits) | 4  8  16 | 270.650 ms  226.700 ms  312.167 ms |
| N\_1=100 si N\_2=100000 (random digits) | 4  8  16 | 29232.4 ms  27487.7 ms  31331.2 ms |
| N\_1=N\_2=16 ; Numar1=”9999 4444 9999 9999” Numar1=”9999 5555 9999 9999” | 4 | 15.9249 ms |

* Varianta 2:

|  |  |  |
| --- | --- | --- |
| Numar 1 = “123456789123456789” = Numar2 | 4  8  16 | 11.9006 ms  7.0508 ms  16.4594 ms |
| N\_1=1000 si N\_2=1000 (random digits) | 4  8  16 | 197.723 ms  157.243 ms  240.600 ms |
| N\_1=100 si N\_2=100000 (random digits) | 4  8  16 | 27148.5 ms  25299.1 ms  29011.3 ms |
| N\_1=N\_2=16 ; Numar1=”9999 4444 9999 9999” Numar1=”9999 5555 9999 9999” | 4 | 14.8301 ms |

Concluzii

**Varianta 0**, cea secvențială, oferă performanțe excelente pentru probleme de dimensiuni reduse, datorită lipsei overhead-ului asociat cu gestionarea mai multor procese. Astfel, pentru seturi mici de date, abordarea secvențială se dovedește a fi cea mai eficientă, evitând complicațiile și timpul suplimentar necesar pentru sincronizare și comunicare între procese.

În cazul **Variantei 1**, timpul de execuție tinde să crească proporțional cu numărul de procese implicate. Acest fenomen sugerează că sistemul întâmpină dificultăți din cauza unui overhead semnificativ generat de gestionarea paralelismului. Resursele suplimentare necesare pentru coordonarea proceselor și comunicarea între ele conduc la o penalizare de performanță, care devine vizibilă pe măsură ce dimensiunea problemei crește.

**Varianta 2** reprezintă un compromis între cele două abordări anterioare. Aceasta oferă un echilibru între performanță și utilizarea resurselor, fiind, în general, mai rapidă decât Varianta 1. Cu toate acestea, nu garantează întotdeauna o îmbunătățire față de Varianta 0, în special atunci când sunt procesate seturi mari de date. În aceste cazuri, overhead-ul suplimentar poate compensa beneficiile paralelizării, făcând ca varianta secvențială să rămână o opțiune competitivă.