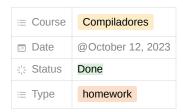
Tabla de operaciones



Threads	Código 1 (SPMD)	Código2 (Pad 8)	Código 3 (Critical)	Código 4 (Atomic)	Código5 (For Reduction)	Código 6 (For Schedule - Static)	Código7 (For Schedule - Dynamic)
1	3.511993	3.507694	3.551492	3.58994	3.481185	3.499105	3.570879
2	1.870533	1.857656	1.841669	1.847546	1.791395	1.785269	5.572519
3	1.272194	1.299302	1.260976	1.270981	1.229863	1.267854	4.085412
4	0.949415	0.981844	0.980947	0.969379	0.933124	0.969389	3.473589
5	1.081764	1.007623	1.121309	1.088127	0.853911	0.853166	3.445156
6	1.096615	0.914924	1.182355	1.092444	0.79481	0.791406	3.252502
7	1.073909	0.780698	1.076815	1.051359	0.755007	0.728365	3.251072
8	1.0947	0.678691	1.110429	1.091778	0.703751	0.690438	3.037108

Numero de pasos: 1,000,000,000

Realizar las gráfica de las tres métricas (speedup, eficiencia y paralelicibilidad) y explicar a manera de concusiones el análisis de tus resultados obtenidos indicando cual fue el mejor de todos.

Tengo dudas sobre cómo he estructurado las tablas siguiendo la fórmula y la información proporcionada, y he llegado a las siguientes conclusiones:

En términos generales, el Schedule Guided parece ser la mejor opción según dos de mis gráficos, aquellos que representan el speedup y la paralelización. Además, el Schedule Auto obtuvo el segundo lugar en términos de datos y gráficos.

Existen varios tipos de programación que ofrecen resultados similares cuando se utilizan más de dos hilos, ya que los primeros siempre mostraron valores mucho más altos.

También es importante destacar que el Schedule Dynamic obtuvo resultados significativamente peores según los datos, y el uso de "critical" arrojó resultados interesantes pero no tan favorables.

Especificaciones la computadora → Macbook Air M1 8 nucleos 8 GB de RAM

Link para spreedsheet con las Tablas y Gráficas

 $https://docs.google.com/spreadsheets/d/1zUWSPB4jzRhPGIA1tm5RUFAU2EPb5_5J7FSriC_0qfA/edit?usp=sharing+ the properties of the properties o$

Tabla de operaciones