

Informe sobre mutex y busy waiting en el calculo de pi

LinkedList

Primero se mostrara la comparaciones que hacen en el libro. Con 100000 operaciones 99.9% member, 0.05%insert, 0,05 % Delete.

Implementation	Number of Threads			
	1	2	4	8
Read-Write Locks	0.213	0.123	0.098	0.115
One Mutex for Entire List	0.211	0.450	0.385	0.457
One Mutex per Node	1.680	5.700	3.450	2.700

Implementación	1	2	4	8
Read- Write Locks	1.08	1.94	2.23	2.21
One Mutex for Entire List	0.094	0.16	0.29	0.32
One Mutex per Node	1.04	3.30	2.40	2.34

En la siguiente tabla se muestra las comparaciones con Read-WriteLocks, OnemutexforEntire ListyOnemutexper NodeSegunda tabla con 100000 operaciones 80% member, 10%insert,10%Delete

Implementation	Number of Threads			
	1	2	4	8
Read-Write Locks	2.48	4.97	4.69	4.71
One Mutex for Entire List	2.50	5.13	5.04	5.11
One Mutex per Node	12.00	29.60	17.00	12.00

Implementación	1	2	4	8
Read- Write Locks	1.34	3.06	3.69	4.39
One Mutex for Entire List	1.02	1.75	2.64	4.63
One Mutex per Node	9.39	23.19	20.50	12.01

Multiplicación de Matriz Vector

En el siguiente cuadro se muestra los resultados del libro.

Threads	Matrix Dimension					
	8,000,000 × 8		8000 × 8000		8 × 8,000,000	
	Time	Eff.	Time	Eff.	Time	Eff.
1	0.393	1.000	0.345	1.000	0.441	1.000
2	0.217	0.906	0.188	0.918	0.300	0.735
4	0.139	0.707	0.115	0.750	0.388	0.290

Implementación	8000000x8	2	8000x8000	8	8x8000000	
Threads	tiempo	efi	tiempo	efi	tiempo	efi
1	0.18	1.00	0.21	1.00	0.20	1.00
2	0.11	0.85	0.12	0.48	0.12	0.23
4	0.09	0.88	0.08	0.61	0.09	0.28