

# Algoritmos Paralelos / Laboratorio

Joseline Maria Zevallos Aliaga  
joseline.zevallos@ucsp.edu.pe  
Universidad Católica San Pablo

## 1 Comparación de Tiempos de BusyWait y Mutex

En la siguiente comparación se tiene el calculo del número de Pi, del cual se esta utilizando Busy waiting, mutex. En la Figura 1 se muestra las comparaciones del libro.

<b>Table 4.1</b> Run-Times (in Seconds) of $\pi$ Programs Using $n = 10^8$ Terms on a System with Two Four-Core Processors		
<b>Threads</b>	<b>Busy-Wait</b>	<b>Mutex</b>
1	2.90	2.90
2	1.45	1.45
4	0.73	0.73
8	0.38	0.38
16	0.50	0.38
32	0.80	0.40
64	3.56	0.38

Figure 1:

En la siguiente tabla se hacen las pruebas usando Busy waiting y Mutex, en los cuales podemos ver que es general, el rendimiento de Mutex es superior en casi todas las pruebas. Eso se debe a que, el uso de Busy Waiting, desperdicia iteraciones esperando a que el recurso se desbloquee para poder hacer el trabajo.

Threads	Busy-Wait	Mutex
1	1.19	1.21
2	0.56	0.82
4	0.50	0.53
8	0.52	0.44
16	0.35	0.41
32	0.34	0.39
64	0.36	0.38

Table 1: Comparación de Busy-Waiting y Mutex

## 2 Linked List

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

Figure 2:

Implementation	1	2	4	8
Read-Write Locks	1.08	1.94	3.23	5.21
One Mutex for Entire List	0.094	0.16	0.29	0.32
One Mutex per Node	1.04	3.30	2.46	2.34

Table 2: Comparación de tiempos con diferentes Threads

En la siguiente tabla se muestra las comparaciones con Read-Write Locks, One mutex for Entire List y One mutex per Node

Segunda 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

Figure 3:

Implementation	1	2	4	8
Read-Write Locks	1.34	2.06	2.69	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

Table 3: Comparación de tiempos con diferentes Threads

### 3 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

Figure 4:

En este cuadro hacemos las pruebas de una multiplicación matriz con un vector.

Threads	8000000x8 Tiempo	Efi	8000x8000 Tiempo	Efi	8x8000000 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
8	0.09	0.83	0.09	0.53	0.08	0.27

Table 4: Multiplicación de matriz-vetor