

Laboratorio 2 - OpenMP

En este laboratorio se hizo la versión secuencial y paralela de Quicksort. En donde se generan 1 000 000 de números aleatorios los cuales son guardados en un archivo y posteriormente se leen para aplicarles cualquiera de las dos versiones, ordenar el arreglo de números aleatorios y guardar estos datos dentro de un nuevo archivo. En donde se calcularon los tiempos para identificar el speed up y eficiencia. Con base a los resultados se observa que hubo una mejora significativa al momento de realizar el programa en paralelo.

Tabla de tiempos:

Número de Hilos		10
	Secuencial	Paralelo 1
Tiempo 1	0.594	0.165
Tiempo 2	0.596	0.169
Tiempo 3	0.593	0.166
Tiempo 4	0.602	0.161
Tiempo 5	0.595	0.167
Promedio	0.596	0.1656
Speedup		3.599034
Eficiencia		0.359903

Screen Shots:

Programa Secuencial:

```
● PS C:\Users\Brand\Desktop\LAB02-PARALELA> .\sort.exe
Tiempo estimado = 0.594
Para N = 1000000
●
PS C:\Users\Brand\Desktop\LAB02-PARALELA> .\sort.exe
Tiempo estimado = 0.596
Para N = 1000000
●
PS C:\Users\Brand\Desktop\LAB02-PARALELA> .\sort.exe
Tiempo estimado = 0.593
Para N = 1000000
●
PS C:\Users\Brand\Desktop\LAB02-PARALELA> .\sort.exe
Tiempo estimado = 0.602
Para N = 1000000
●
PS C:\Users\Brand\Desktop\LAB02-PARALELA> .\sort.exe
Tiempo estimado = 0.595
Para N = 1000000
```

Programa Paralelo:

```
● PS C:\Users\Brand\Desktop\LAB02-PARALELA> .\psort.exe
Tiempo estimado = 0.165
Para N = 1000000
Para n_threads = 10
● PS C:\Users\Brand\Desktop\LAB02-PARALELA> .\psort.exe
Tiempo estimado = 0.169
Para N = 1000000
Para n_threads = 10
● PS C:\Users\Brand\Desktop\LAB02-PARALELA> .\psort.exe
Tiempo estimado = 0.166
Para N = 1000000
Para n_threads = 10
● PS C:\Users\Brand\Desktop\LAB02-PARALELA> .\psort.exe
Tiempo estimado = 0.161
Para N = 1000000
Para n_threads = 10
● PS C:\Users\Brand\Desktop\LAB02-PARALELA> .\psort.exe
Tiempo estimado = 0.167
Para N = 1000000
Para n_threads = 10
○ PS C:\Users\Brand\Desktop\LAB02-PARALELA> █
```