



**Tecnológico
de Monterrey**

Campus Puebla

Multiprocesadores

Actividad 1.5

Jorge Martínez Hernández – A01731762

Profesor: Dr. Emmanuel Torres Rios

Introducción

En esta actividad se busca comparar tiempos entre la memoria RAM y la cache de mi equipo, utilizando algoritmos vistos en clase. Buscando identificar y deducir el porqué de cada tiempo.

Objetivo

Con los programas analizados durante clase obtendremos tres puntos importantes:

- 1) Velocidad de lecto-escritura de la memoria RAM de su equipo mediante threads en comparación con la cache
- 2) Velocidad de lecto-escritura del disco duro de su equipo mediante threads en comparación con la cache
- 3) La operación matemática (exponencial, potencia, trigonométrica, etc.) más lenta en comparación con la suma.

Desarrollo

Ejercicio 1:

Se ejecutaron los programas Euler_array.c ya que este programa muestra el tiempo que le toma a la RAM y parallel.c y el segundo nos muestra el tiempo en la memoria cache. Es de suma importancia encontrar los parámetros óptimos con los que estos programas funcionan y ejecutan sus funciones de manera correcta. Los parámetros que tenemos que enfocarnos es el numero de threads y el número de pasos.

```
C:\Users\Jorge MH\Documents\tec\Onceavo Semestre\Multiprocesadores>gcc -fopenmp euler_array1.c  
C:\Users\Jorge MH\Documents\tec\Onceavo Semestre\Multiprocesadores>a.exe  
Numero de pasos:120000  
tomo (0.001000) segundos
```

Para esta ejecución se puede observar la respuesta por parte de la memoria RAM

```
C:\Users\Jorge MH\Documents\tec\Onceavo Semestre\Multiprocesadores>gcc -fopenmp pi_parallel2.c  
C:\Users\Jorge MH\Documents\tec\Onceavo Semestre\Multiprocesadores>a.exe  
pi = (3.141593)  
tomo (2.387000) segundos
```

Para esta ejecución se puede observar la respuesta por parte de la memoria cache. Logrando observar que la RAM es mucho más rápido que la cache.

Ejercicio 2:

Para esta parte de esta tarea se comparo ahora el tiempo de la memoria cache y el tiempo del disco duro. De igual manera se consideraron parámetros para poder correr los programas, en este caso como se asignaron 6 threads los pasos que tenemos en el programa que mide el tiempo de la cache los dividimos entre el número de threads con los que se trabajaran para medir el rendimiento del disco duro.

```
C:\Users\Jorge MH\Documents\tec\Onceavo Semestre\Multiprocesadores>gcc -fopenmp euler1.c

C:\Users\Jorge MH\Documents\tec\Onceavo Semestre\Multiprocesadores>a.exe
Numero de pasos:200000 Atendido por thread:3
Numero de pasos:200000 Atendido por thread:5
Numero de pasos:200000 Atendido por thread:4
Numero de pasos:200000 Atendido por thread:0
Numero de pasos:200000 Atendido por thread:1
Numero de pasos:200000 Atendido por thread:2
tomo (0.620000) segundos
```

En esta imagen podemos observar el rendimiento del disco duro, utilizando el programa euler1.c el cual nos ayuda a medir este rendimiento.

```
Numero de pasos:200000 Atendido por thread:2
tomo (0.620000) segundos

C:\Users\Jorge MH\Documents\tec\Onceavo Semestre\Multiprocesadores>gcc -fopenmp pi_parallel12.c

C:\Users\Jorge MH\Documents\tec\Onceavo Semestre\Multiprocesadores>a.exe
pi = (3.141593)
tomo (2.394000) segundos
```

En esta imagen podemos observar el rendimiento de la memoria cache, utilizando el programa pi_parallel1.c el cual nos ayuda a medir este rendimiento.

Ejercicio 3:

Para esta parte de la tarea se solicitó poner alguna operación matemática en este caso yo utilice una exponencial teniendo como tiempo el siguiente:

```
C:\Users\Jorge MH\Documents\tec\Onceavo Semestre\Multiprocesadores>a.exe
Numero de pasos:120000
tomo (0.002000) segundos
```

Conclusión

Podemos concluir que la memoria RAM es la más eficiente, posteriormente el disco duro y finalmente la memoria cache. Teniendo en consideración que se tomaron los mismos parámetros para los tres casos.