

Informe: Caches and Programs

Sol Morelia Velásquez Rodríguez
Departamento de Ciencia de la Computación
Universidad Católica San Pablo
Arequipa - Perú
Email: sol.velasquez@ucsp.edu.pe

1. Descripción

En el programa se presentan una matriz cuadrada A , dos arrays x , y , todos de tipo double. Lo que se desea realizar es guardar en cada elemento de y la sumatoria de los elementos de cada fila de A multiplicados por los elementos de x . Por lo que se proponen dos bucles, que tienen la misma complejidad algorítmica $O(n^2)$. La diferencia entre ellas es el acceso a la matriz A y al array y . En el primer bucle se accede a los valores de A por filas y se mantiene el mismo índice para y hasta terminar de recorrer toda la fila. En el segundo bucle se accede a los valores de A por columnas y se cambia el índice en y para guardar correctamente la sumatoria de las filas. [1]

2. Comparación

Para realizar la comparación se utilizó la variable MAX con los siguientes valores 10 , 100 , 1000 y 10000 . Se tomó el tiempo en segundos con la librería *chrono*. Dando como resultado la siguiente tabla que muestra el tiempo en segundos.

TABLE 1. TABLA QUE MUESTRA LOS TIEMPOS PARA CADA BUCLE

Variable MAX	First Loop	Second Loop
10	0.0000025	0.000001
100	0.000039	0.0000474
1000	0.0031344	0.0043709
10000	0.327767	1.14602

3. Conclusiones

Con todo lo visto anteriormente se puede concluir que el acceso a la caché afecta en el rendimiento de los programas que se realizan, en este caso se vió el principio de localidad espacial. Debido a que en el primer bucle se accedía por filas se evitaban caché misses, ya que en la memoria las matrices son almacenadas como arrays, donde cada fila va una seguida de la otra.

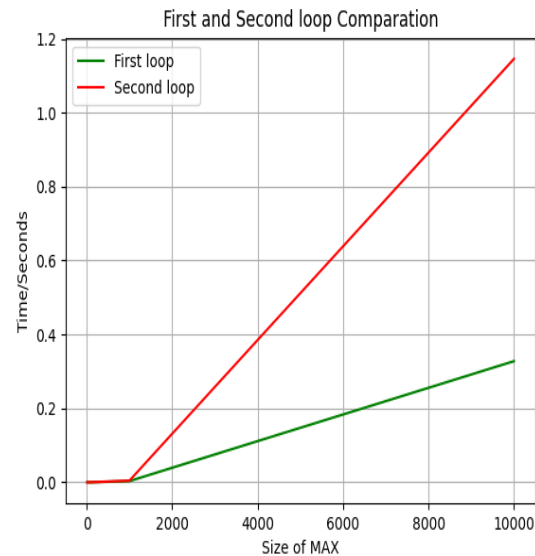


Figure 1. Comparación entre los dos bucles

References

- [1] Solcito25, "Distributed-and-parallel-computing," <https://github.com/Solcito25/Distributed-and-parallel-computing/tree/main/Cache>, 2022.