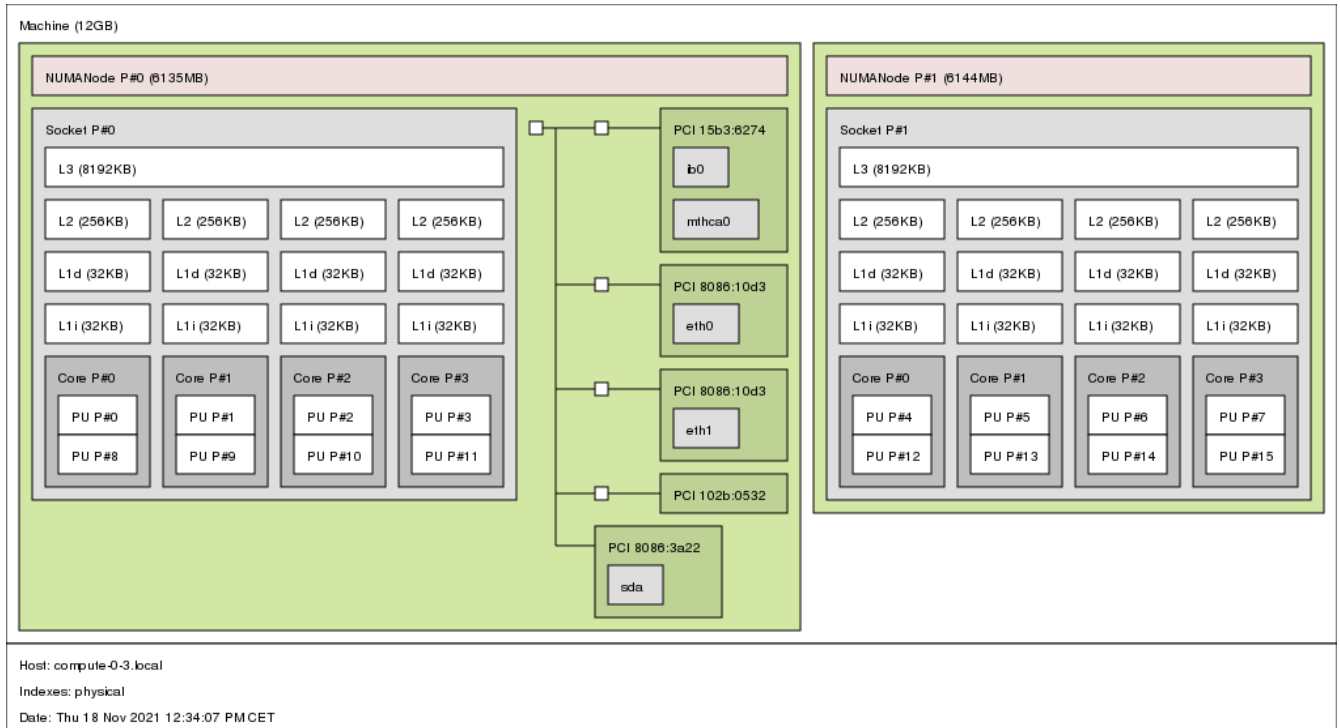


Arquitectura de ordenadores práctica 3: Memoria caché y rendimiento

Ejercicio 0

A continuación se muestra la arquitectura de las memorias caché del procesador del ordenador de uno de los componentes de la pareja.



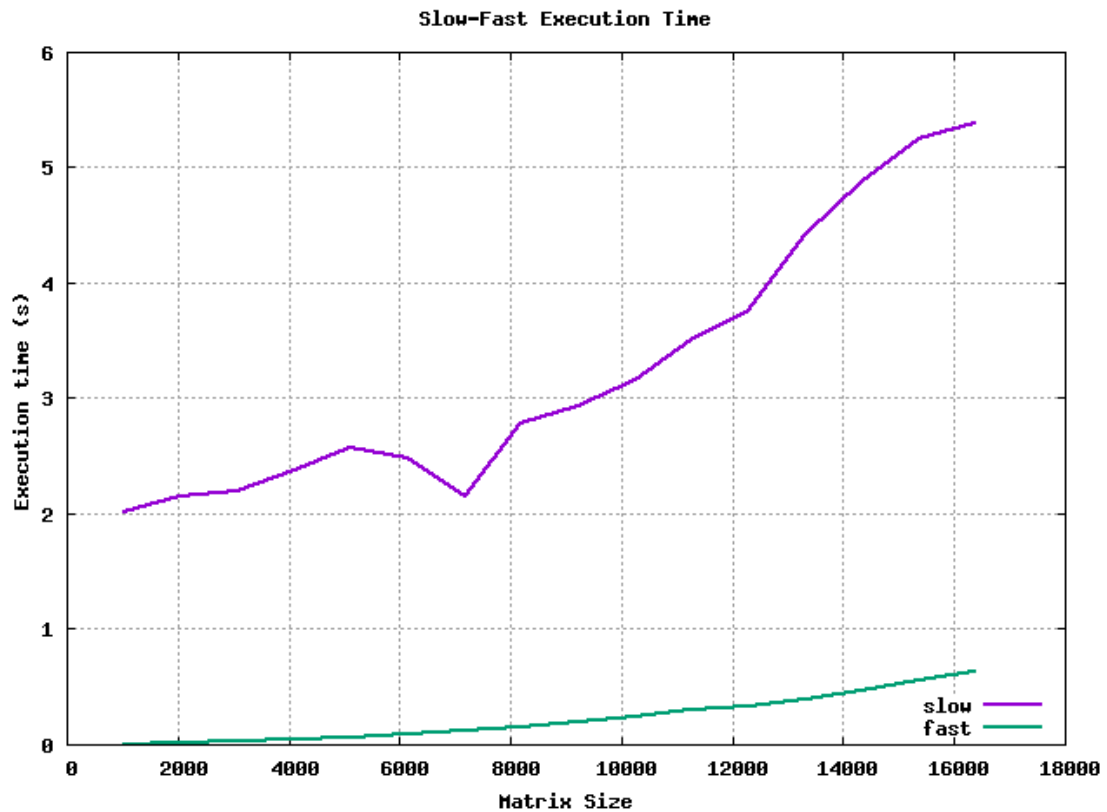
Ejercicio 1

Es necesario realizar múltiples mediciones para cada tamaño de la matriz, porque al medir el rendimiento, la CPU no solo está ejecutando el programa que medimos, sino también ejecutando otros procesos que afectan el plan de ejecución del sistema operativo. Es por ello que al promediar el número de veces que hemos obtenido, podemos en gran medida obtener resultados reales, muy cercanos al tiempo requerido para la ejecución del programa.

Para obtener los datos, se siguieron las sugerencias del enunciado, intercalando el tamaño de la matriz y realizando un número suficiente de iteraciones, y finalmente obteniendo el promedio del tiempo de ejecución. Específicamente, se han utilizado 13 iteraciones, evitando así grandes picos en la medición en la medida de lo posible. Aunque en la medición de slow hay un ligero pico.

La figura no muestra el efecto de que el tiempo de ejecución de los dos algoritmos al principio es similar. Esto se debe principalmente a que estos algoritmos presentan tiempos similares a los tamaños de matriz que no se reflejan en la figura, y no han calculado su

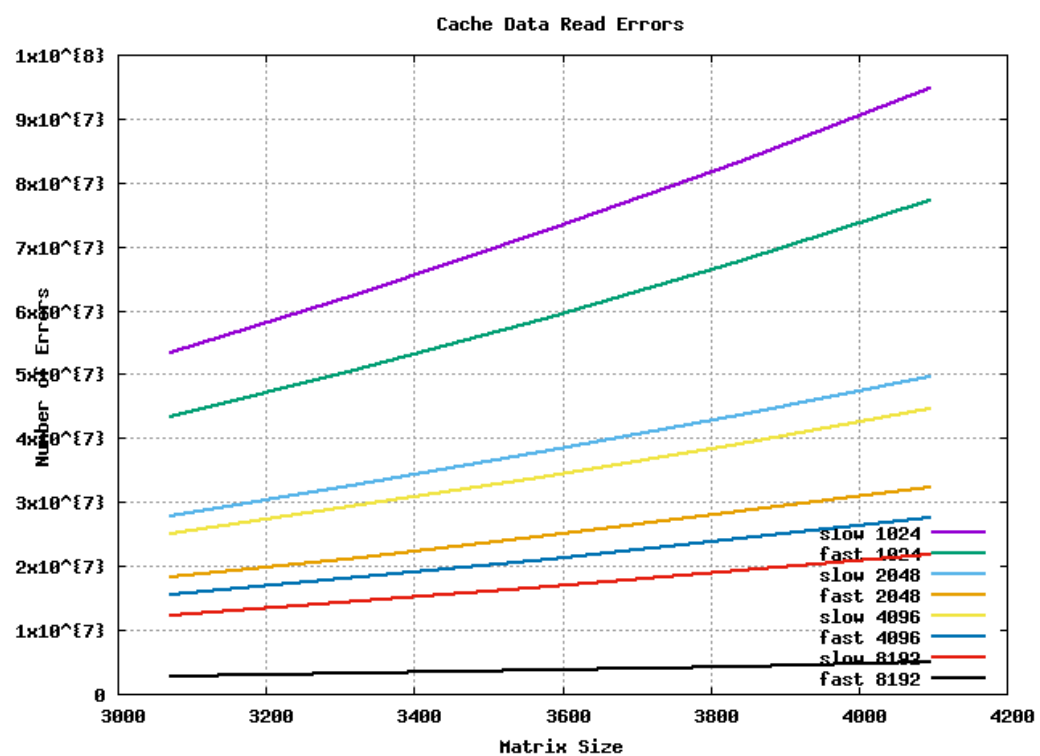
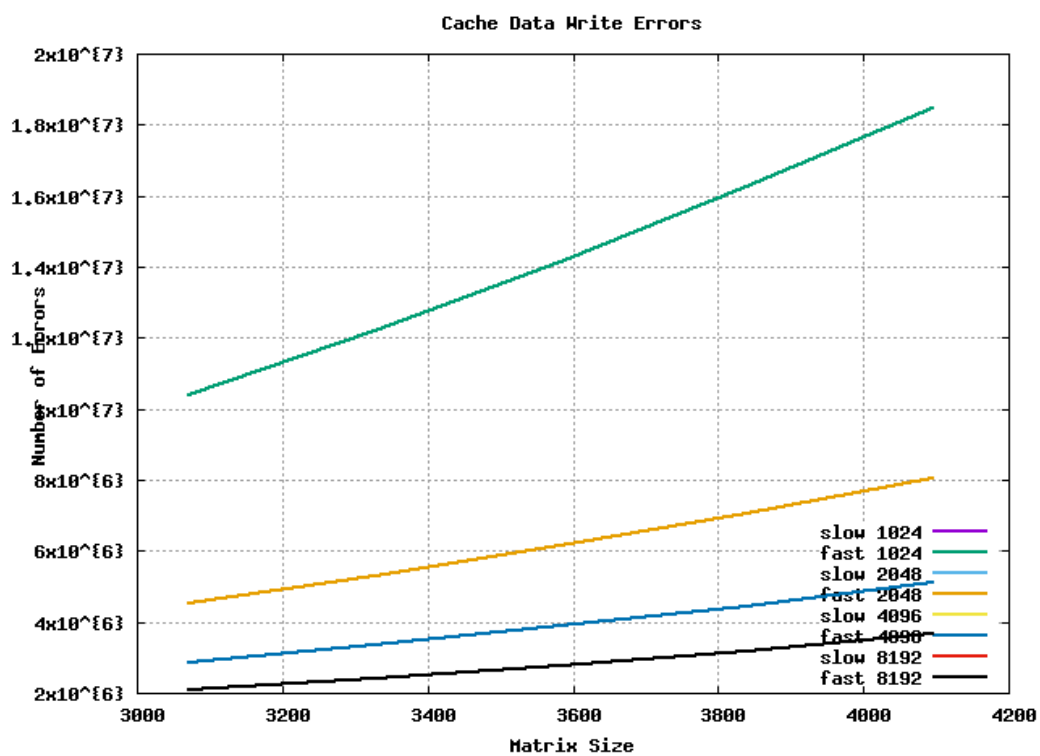
tiempo de ejecución. La diferencia observada entre el algoritmo slow y el algoritmo fast es que el primero accede a los elementos de la matriz por columnas y el segundo accede por filas. Esto afectará el tiempo de ejecución de los dos algoritmos, porque la caché se beneficia de la ubicación espacial de los datos y porque la matriz se almacena en la memoria principal en filas (los elementos están almacenados consecutivamente), cuando un elemento de la matriz se almacena en caché, se incluye el bloque de memoria de esa fila.



Gráfica del ejercicio 1

Ejercicio 2

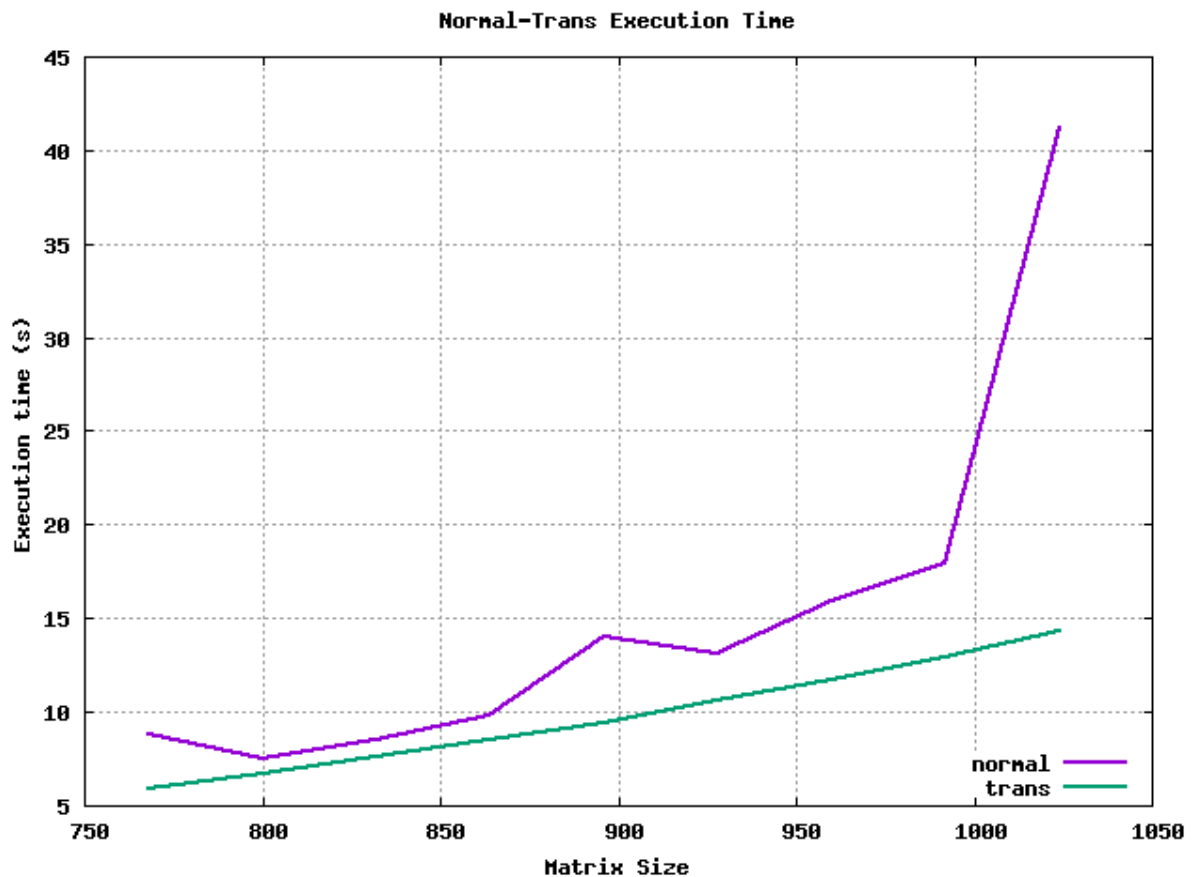
En estas dos gráficas observamos que a medida que aumenta el tamaño de la matriz de los dos programas, el número de errores de caché aumenta significativamente. Esto se debe a que cuanto mayor sea el tamaño de la matriz, más datos se deben leer y escribir en la memoria caché. Además, para el mismo tamaño de matriz, observamos en ambas figuras que cuanto mayor es el tamaño de la caché, menor es el número de fallos. Esto se debe a que se pueden colocar más bloques de memoria en la caché y se reemplazan menos bloques de memoria. Finalmente, debe tenerse en cuenta que para el mismo tamaño de caché, el gráfico de errores de escritura muestra el mismo número tanto para fast como slow. Esto se debe a que el acceso a los elementos escritos en los resultados de los dos algoritmos no depende de ellos, por lo que el número de veces que escribe los elementos está relacionado con el tamaño de la matriz, más que con el algoritmo.

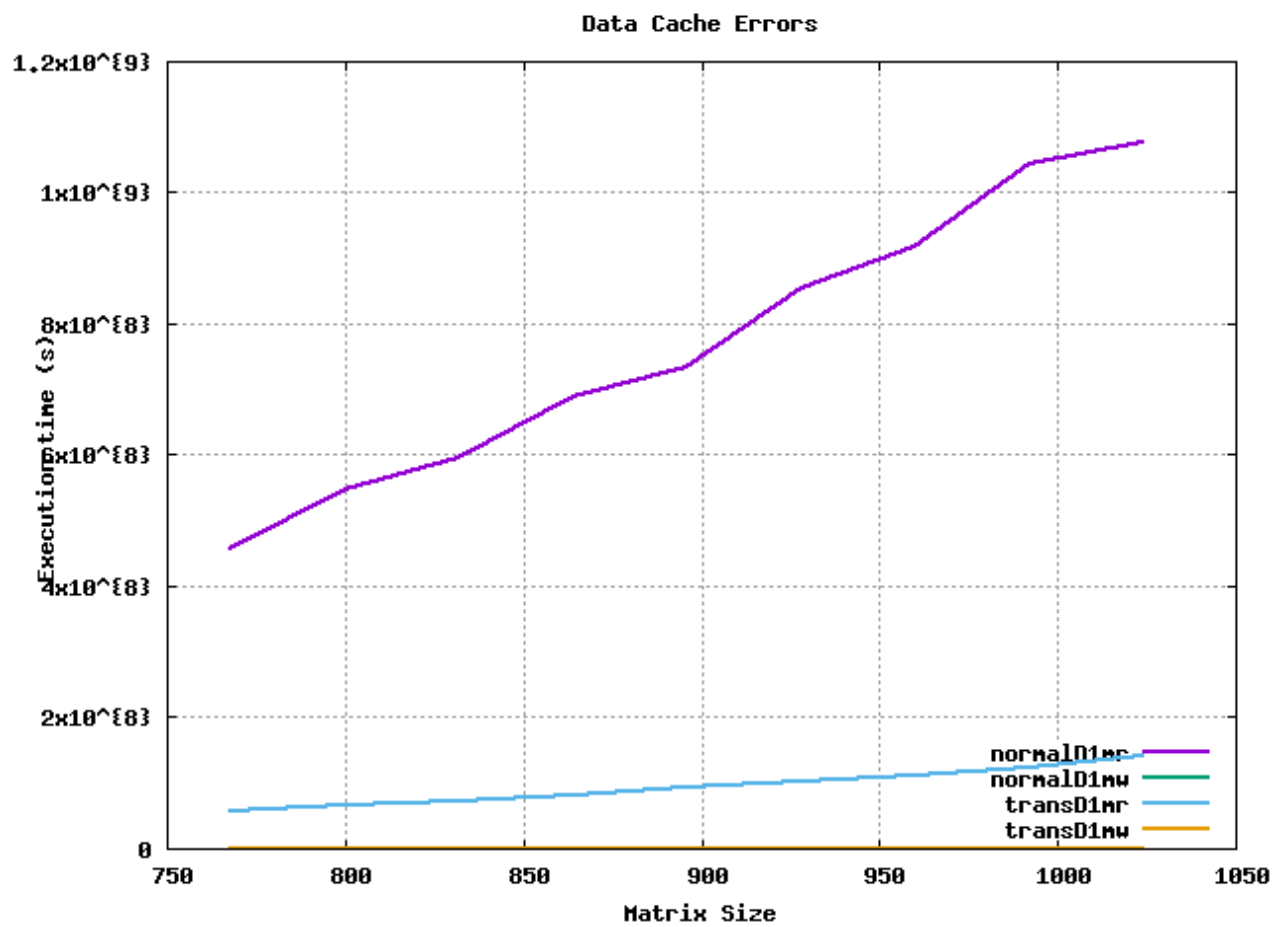


Gráficas del ejercicio 2

Ejercicio 3

Estas gráficas muestran la diferencia clara en el tiempo de ejecución y los errores de caché al calcular la multiplicación normal y la multiplicación transpuesta. Esto se debe principalmente a la influencia del acceso a las filas y columnas de la matriz explicada anteriormente. Específicamente, en la multiplicación normal, se accede a la segunda matriz por columna, lo que significa que debido a la localidad espacial de los datos, hay más errores de caché. Por otro lado, en la multiplicación con transposición, se accede a la segunda matriz por fila, lo que se beneficia de la existencia de datos continuos en la memoria caché, ya que estos datos tienen más probabilidades de encontrarse en el mismo bloque que se transfiere de la memoria principal a la memoria caché.





Gráficas del ejercicio 3

Autores

Bernardo Andrés Zambrano

Luis Miguel Nucifora

Grupo 1311

Nº de pareja 2