Informe de Prácticas

Sesión 1

Computación de Altas Prestaciones

Máster en Ingeniería Informática

Universidad de Oviedo

Autor:

Rubén Martínez Ginzo, UO282651@uniovi.es

Abril 2025

Índice

Índice de figuras

Índice de tablas

1. Introducción

En esta sesión de prácticas de laboratorio se evalúan distintos esquemas de acceso a memoria aplicados a la multiplicación de matrices. Concretamente, se analiza el rendimiento de esta operación matemática utilizando tres estrategias de acceso diferentes:

- *Row-Major order*
- *Column-Major order*
- *Z-Order*

El objetivo principal es determinar cuál de estas estrategias ofrece un mejor desempeño en términos de tiempo de ejecución y eficiencia en el uso de memoria.

1.1. Desarrollo

Para el desarrollo de esta práctica, se han seguido las indicaciones recogidas en el guion de la sesión correspondiente. Se han implementado las tres estrategias de acceso en el código fuente, y se han llevado a cabo diversas pruebas para medir su rendimiento. Con el fin de comparar los tiempos de ejecución, se ha desarrollado un programa en lenguaje C con fines de *benchmarking*, tal y como se detalla en el capítulo siguiente. Todo el código fuente se encuentra disponible públicamente en el siguiente repositorio de GitHub, así como en el archivo *zip* asociado a esta entrega.

1.2. *Benchmarking*

Además del trabajo realizado durante la sesión de laboratorio, se ha desarrollado un programa en C que automatiza la medición de los tiempos de ejecución para las distintas estrategias de acceso a memoria.

Este programa, ubicado en el subdirectorio *benchmark*, permite especificar diferentes tamaños de matriz y un número fijo de iteraciones para cada experimento. Durante la ejecución, se realiza la multiplicación de matrices utilizando cada uno de los esquemas de acceso, registrando el tiempo requerido por cada estrategia. En el caso particular del algoritmo *zorder*, se evalúan automáticamente todos los tamaños de bloque posibles para cada tamaño de matriz, llevando a cabo la multiplicación con cada configuración. Cada multiplicación se repite tantas veces como iteraciones se hayan especificado, calculando el tiempo medio de ejecución de todas ellas. Esto permite reducir significativamente el ruido en las mediciones y obtener resultados más precisos.

Al finalizar, el programa devuelve un *log* en formato CSV con los resultados otenidos. Dichos tiempos de ejecución constituyen la base de los análisis y comparativas que se presentan en los capítulos siguientes.

2. *Row-Major order*

3. *Column-Major order*

4. *Z-Order*

5. Comparativa