

Taller evaluación de Rendimiento



Profesionales en Formación

Daniela Moreno Leon

Miguel Duarte Clavijo

Pontificia Universidad Javeriana

Facultad de ingeniería

Sistemas Operativos

Bogota D.C.

2024

Índice

0.- Introducción

1.- Contexto de Sistemas Operativos

2.- Comparativa de Sistemas de cómputo (Plataformas de cómputo seleccionadas)

3.- Comparativa a través de Algoritmos de MM (cúal), implementados en C(porque)

4.- Paradigmas de Programación: Serie / Paralelo

5.- Metodología de experimentación (basado en Ley de Grandes Números)

6.- Análisis Exhaustivo de los datos

6.1- Repositorio de codigo(GitHub)

7.-Resumen

8.- Conclusiones

9.- Referencias

0.- Introducción

En el contexto del Taller de Evaluación de Rendimiento, se ha llevado a cabo un análisis exhaustivo para medir y comparar el rendimiento de diferentes programas. Este informe presenta los resultados de dicho análisis, centrándose en dos aspectos fundamentales: las métricas de rendimiento utilizadas y las plataformas de hardware y software empleadas para realizar las mediciones.

Las métricas de rendimiento desempeñan un papel crucial en la evaluación de la eficiencia y optimización de los programas. En este taller, nos enfocamos principalmente en el tiempo de ejecución, medida en microsegundos (μs), como una métrica fundamental para evaluar la eficiencia temporal de los programas bajo diferentes condiciones y tamaños de entrada. Además, se discuten otras métricas comúnmente utilizadas en la evaluación de rendimiento y se explican sus implicaciones en el análisis de programas.

Se iniciará dando un breve contexto sobre los sistemas operativos con el fin de que se logre tener un contexto más amplio de lo que se desea realizar. Continuando se hará una comparativa de los sistemas de cómputo seleccionados, estos fueron Windows y Linux, ya que la solución del taller se basó en encontrar qué sistema de cómputo es más eficiente a la hora de usarlo. A partir de aquí se partirá del algoritmo de la multiplicación de matriz normal y transpuesta para comparar su rapidez a la hora de la utilización de hilos. Para seguir comprendiendo se describirán diferentes paradigmas de programación en serie y en paralelo, los cuales cumplen funciones diferentes. Y finalizando se realizará la experimentación, un analisis exhaustivo de los datos resultantes y se terminará con una conclusión.

El presente informe busca proporcionar una visión completa y detallada del proceso de evaluación de rendimiento realizado en el taller, así como de los resultados obtenidos. Esto

permitirá comprender mejor el rendimiento de los programas analizados y facilitará la identificación de posibles áreas de mejora y optimización

1.- Contexto de Sistemas Operativos

Un Sistema Operativo (SO) es un software diseñado para controlar el hardware de un sistema de procesamiento de datos específico, con el fin de permitir que los usuarios y los programas de aplicación lo utilicen. Es el software que controla la asignación y el uso de recursos de hardware como la memoria, el tiempo de CPU, el espacio en disco y los dispositivos de entrada y salida. En términos de ciencias de la computación, es el software que controla la ejecución de programas de computadora y puede proporcionar diversos servicios. En resumen, un sistema operativo es esencial para el funcionamiento de cualquier dispositivo informático, ya que actúa como un intermediario entre el hardware y el usuario, gestionando los recursos del sistema y permitiendo la ejecución eficiente de programas y aplicaciones.

2.- Comparativa de Sistemas de cómputo (Plataformas de cómputo seleccionadas)

Linux y Windows son dos sistemas operativos muy populares, cada uno con sus propias fortalezas y debilidades. Aunque ambos tienen el mismo objetivo principal, que es permitir la interacción entre el usuario y el hardware de la computadora, difieren en muchos aspectos.

Linux es un sistema operativo de código abierto, lo que significa que cualquier persona puede ver, modificar y contribuir al código fuente. Esto ha llevado a una gran cantidad de distribuciones de Linux, cada una adaptada a diferentes tipos de usuarios y necesidades. Linux es conocido por su estabilidad y eficiencia, y es ampliamente utilizado en servidores y sistemas embebidos. Además, Linux es altamente personalizable, lo que permite a los usuarios ajustar el sistema a sus necesidades específicas. Sin embargo, esta personalización puede requerir un conocimiento técnico más profundo, lo que puede ser un desafío para los usuarios menos técnicos.

Por otro lado, Windows es un sistema operativo de código cerrado desarrollado por Microsoft. Esto significa que su código fuente no está disponible para el público, por lo que los usuarios y desarrolladores no pueden modificarlo. Windows es conocido por su interfaz de usuario amigable, que es fácil de entender y usar incluso para los usuarios no técnicos. Además, Windows tiene una gran base de usuarios y una amplia compatibilidad con software y hardware. Sin embargo, Windows puede ser menos eficiente en términos de uso de recursos del sistema en comparación con Linux, y es más susceptible a virus y malware debido a su popularidad.

En resumen, la elección entre Linux y Windows depende en gran medida de las necesidades y habilidades del usuario. Linux puede ser una excelente opción para aquellos que valoran la personalización, la eficiencia y la estabilidad, y no les importa tener que aprender más sobre el funcionamiento de su sistema operativo. Por otro lado, Windows puede ser más adecuado

para aquellos que buscan una experiencia de usuario sencilla y una amplia compatibilidad con diversas aplicaciones y dispositivos.

3.- Comparativa a través de Algoritmos de MM (cúal), implementados en C(porque)

Ambos códigos realizan la multiplicación de matrices utilizando múltiples hilos, pero la principal diferencia radica en cómo acceden a los elementos de las matrices durante la multiplicación. En la multiplicación de matrices clásico la multiplicación se realiza de la manera clásica, es decir, cada elemento de la matriz resultante se calcula como la suma de los productos de los elementos correspondientes de las filas de la primera matriz y las columnas de la segunda matriz. Por otro lado, en la multiplicación de matrices transpuestas, se utiliza una versión transpuesta de la multiplicación de matrices. En esta versión, la segunda matriz se transpone primero, lo que significa que sus filas y columnas se intercambian. Luego, la multiplicación se realiza de la misma manera que en el método clásico, pero ahora con la matriz transpuesta. Esto puede ser más eficiente en términos de rendimiento debido a cómo se almacenan y acceden los elementos de la matriz en la memoria.

4.- Paradigmas de Programación: Serie / Paralelo

Los paradigmas de programación en serie y en paralelo se refieren a dos enfoques distintos para la ejecución de tareas en la programación.

La programación en serie explica que en este paradigma, las tareas se ejecutan de manera secuencial, es decir, una tarea debe completarse antes de que comience la siguiente. Esto es común en la mayoría de los programas que ejecutamos en nuestros ordenadores, donde las instrucciones se ejecutan una tras otra.

La programación en paralelo en diferencia a la programación en paralelo implica ejecutar múltiples tareas o sub-tareas simultáneamente. Esto se logra dividiendo un problema en partes más pequeñas que pueden ser resueltas al mismo tiempo. Este enfoque es especialmente útil en sistemas con múltiples procesadores o núcleos, ya que cada uno puede trabajar en una parte del problema, lo que puede resultar en una ejecución más rápida del programa.

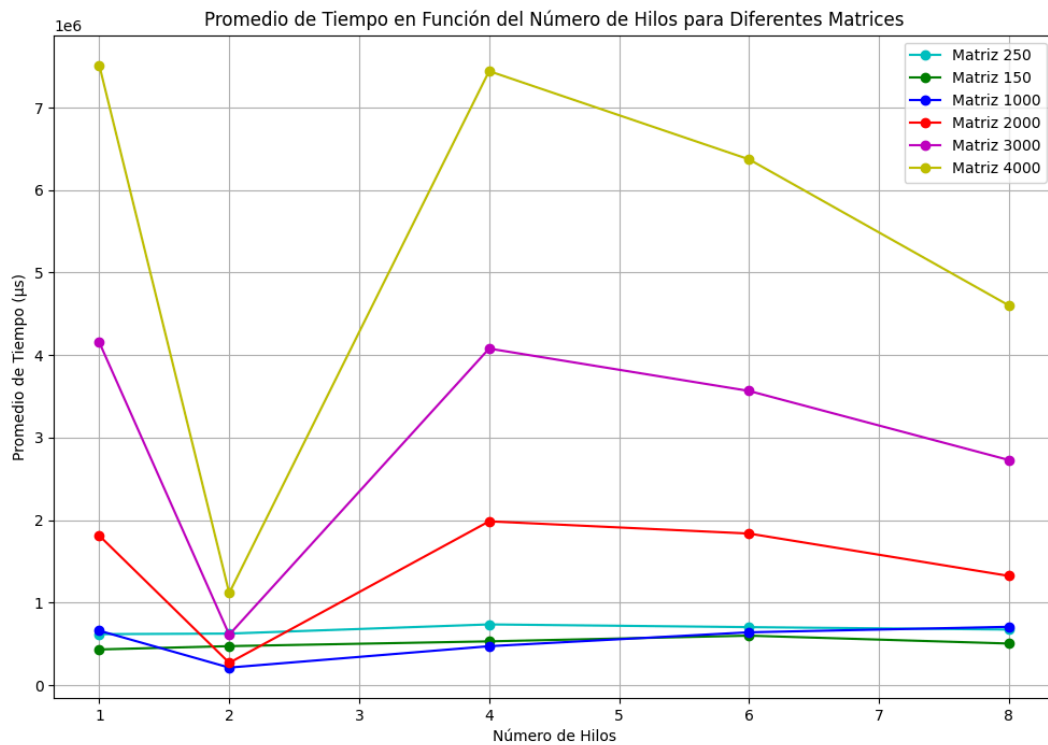
Es importante mencionar que aunque la programación en paralelo puede aumentar la eficiencia, también puede aumentar la complejidad del código. Coordinar múltiples tareas puede requerir sincronización y comunicación entre tareas, lo que puede ser un desafío. Además, no todos los problemas se prestan a una solución paralela eficiente. En algunos casos, la programación en serie puede ser la opción más adecuada.

5.- Metodología de experimentación (basado en Ley de Grandes Números)

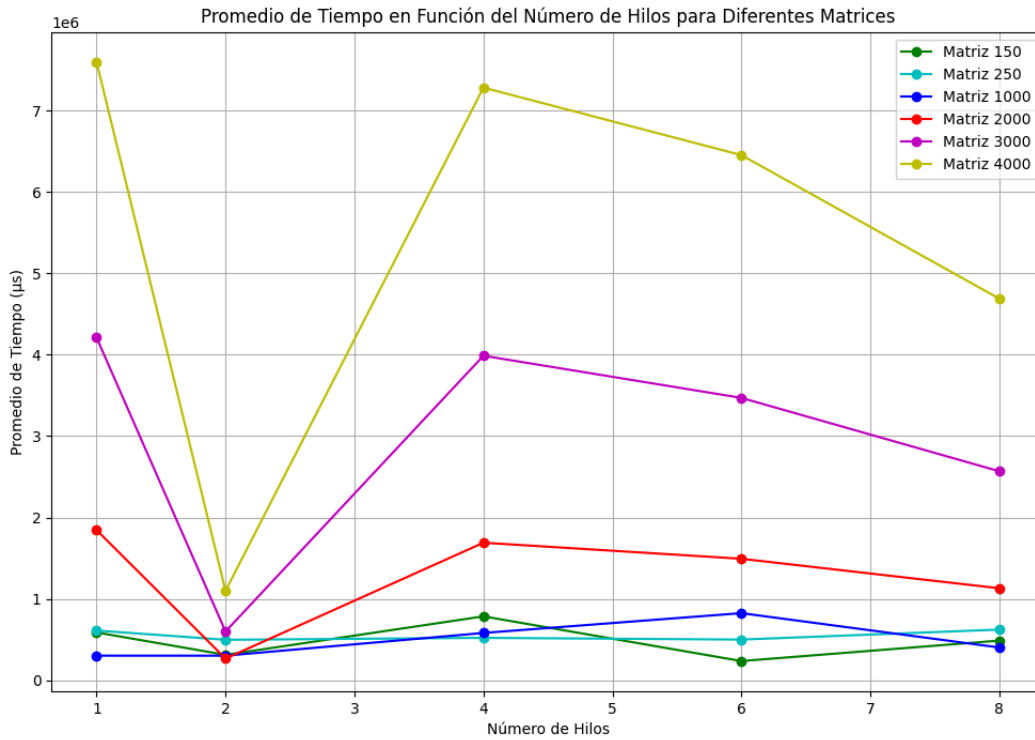
La Ley de los Grandes Números es un teorema fundamental en la teoría de la probabilidad. Esta ley establece que si se repite un experimento aleatorio bajo las mismas condiciones un número grande de veces, el promedio de los resultados tiende a acercarse al valor esperado.

La multiplicación de matrices de tamaños grandes es una operación algebraica que se realiza de acuerdo con las reglas de la multiplicación de matrices. El costo computacional de esta operación depende del tamaño de las matrices involucradas y el algoritmo utilizado para realizar la multiplicación. Cuando las matrices son de tamaños grandes, la multiplicación puede ser una operación costosa en términos de tiempo y recursos computacionales.

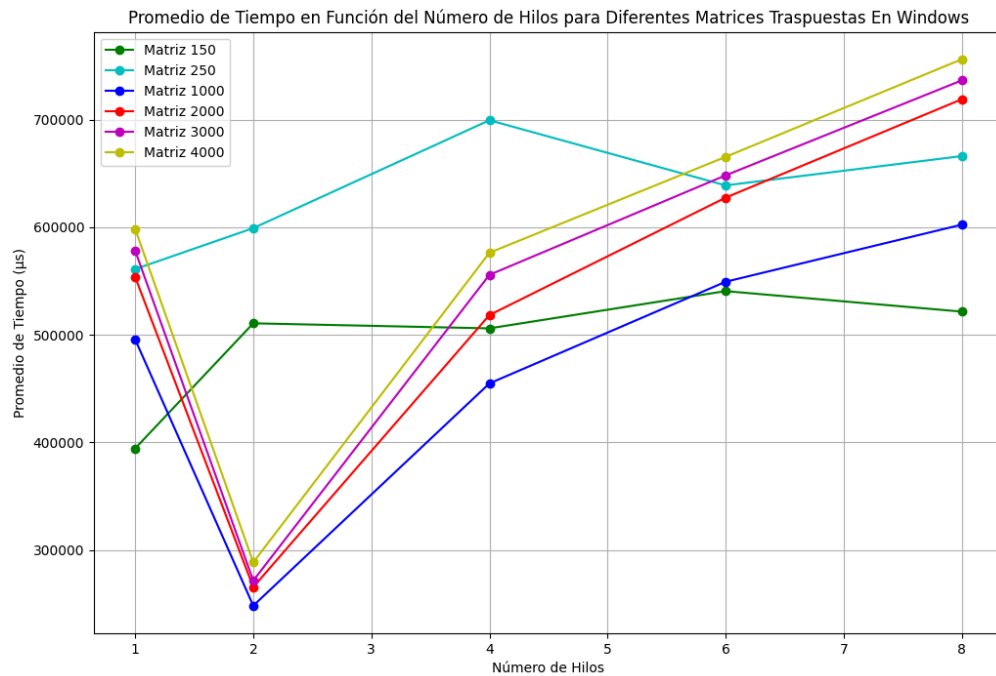
6.- Análisis Exhaustivo de los datos



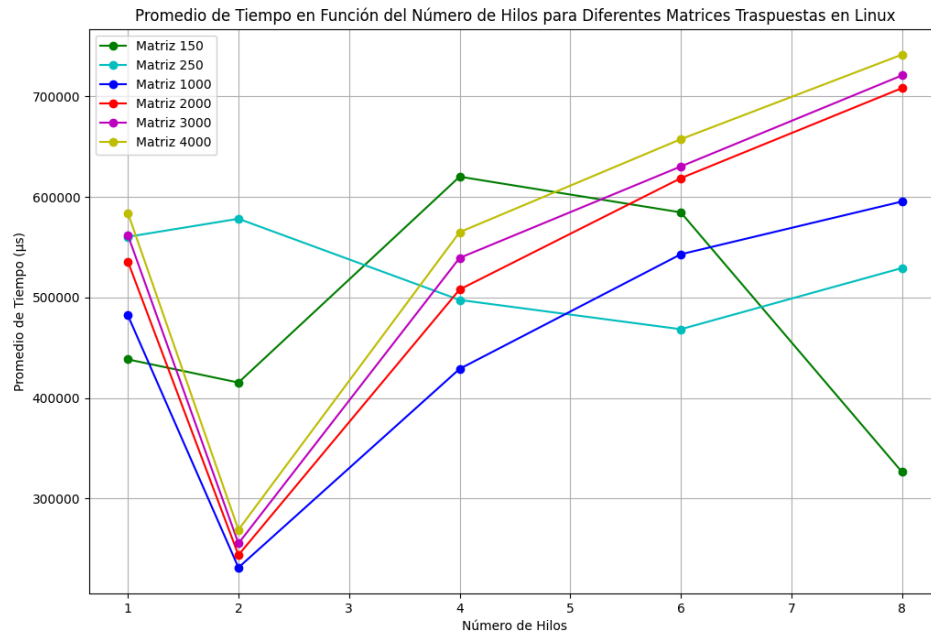
*Gráfica de Promedio de Tiempo por números de hilos en MM_clasico en el Sistema Operativo Windows
(Gráfica 1).*



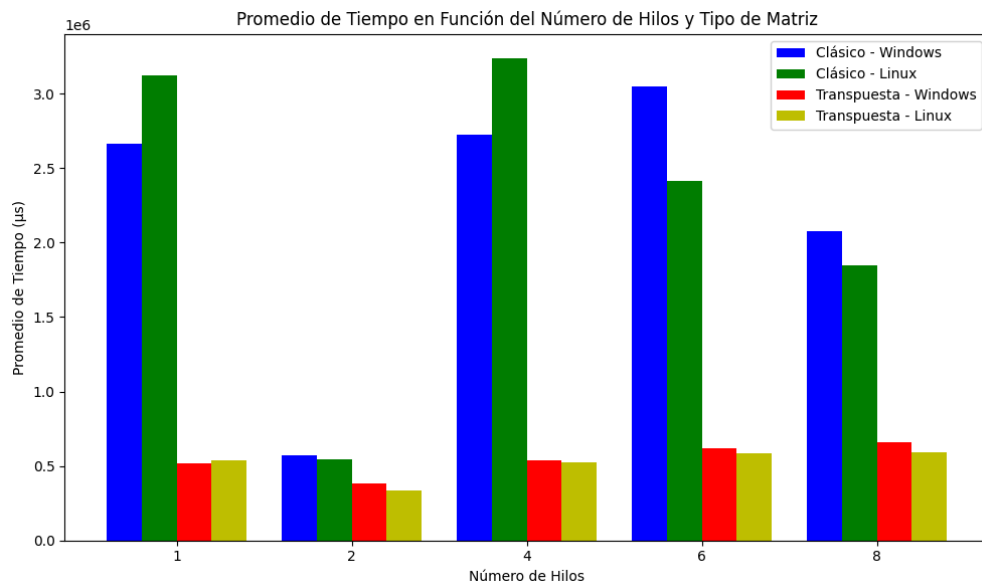
Gráfica de Promedio de Tiempo por números de hilos en MM_clasico en el Sistema Operativo Linux (Gráfica 2).



Gráfica de Promedio de Tiempo por números de hilos en MM_Transpuesta en el Sistema Operativo Windows (Gráfica 3).



Gráfica de Promedio de Tiempo por números de hilos en MM_Transpuesta en el Sistema Operativo Linux (Gráfica 4).



Gráfica del comportamiento de los 2 tipos de matrices en los diferentes sistemas operativos (Gráfica 5).

Se realizaron experimentos de multiplicación de matrices utilizando dos enfoques: multiplicación normal y multiplicación con la matriz transpuesta. Se evaluaron matrices de diferentes tamaños (150x150, 250x250, 1000x1000, 2000x2000, 3000x3000 y 4000x4000) y se varió el número de hilos de ejecución (1, 2, 4, 6 y 8). Los experimentos se llevaron a cabo en dos sistemas operativos, Windows y Ubuntu(Docker).

Resultados Principales

1. Rendimiento de la Multiplicación Transpuesta: Se observó una mejora en el rendimiento al utilizar la multiplicación con la matriz transpuesta en comparación con la multiplicación normal. Esta mejora se hizo más evidente a medida que aumentaba el tamaño de la matriz.
2. Impacto de los Hilos de Ejecución: El uso de múltiples hilos (pthreads) resultó en una mejora significativa en el rendimiento de ambos métodos de multiplicación, especialmente para matrices de tamaños más grandes. Sin embargo, esta mejora se estabilizó después de un cierto número de hilos, sugiriendo un punto de saturación en el que el beneficio adicional de añadir más hilos se vio contrarrestado por la sobrecarga de la gestión de procesos concurrentes.
3. Diferencias entre Sistemas Operativos: No se observaron diferencias significativas en el rendimiento entre Windows y Linux. Ambos sistemas mostraron un comportamiento similar en la ejecución de las operaciones de multiplicación de matrices, lo que sugiere una consistencia en el rendimiento independientemente del sistema operativo utilizado.

Interpretación

- La ventaja de la multiplicación con la matriz transpuesta radica en su capacidad para aprovechar mejor la localidad espacial de los datos, lo que resulta en un acceso más eficiente a la memoria caché.
- El uso de hilos permite una mejor utilización de los recursos de procesamiento disponibles en sistemas multi-core, lo que conduce a una mejora en el rendimiento, especialmente para operaciones con matrices de gran tamaño.
- La estabilización del rendimiento después de cierto número de hilos sugiere que es importante encontrar un equilibrio entre el paralelismo y la sobrecarga de la gestión de hilos.
- La consistencia en el rendimiento entre diferentes sistemas operativos resalta la portabilidad y la robustez de los algoritmos implementados.
- Se recomienda el uso de la multiplicación con la matriz transpuesta para aplicaciones que trabajen con matrices de gran tamaño, ya que ofrece un rendimiento óptimo en comparación con la multiplicación normal.

Este análisis exhaustivo proporciona una comprensión detallada del rendimiento de los algoritmos de multiplicación de matrices en diferentes condiciones y entornos de ejecución.

6.1-Repositorio de código

[Enlace hacia el repositorio de github donde se encuentra el código completo con sus respectivos comentarios explicandolo](#)

7.- Resumen

- Problemas Frente a los Sistemas de Cómputo

Los sistemas de cómputo, representados por los sistemas operativos Windows y Linux, presentan varios desafíos que afectan su rendimiento y eficiencia. Windows es conocido por su facilidad de uso y amplia compatibilidad con software y hardware, lo que lo hace accesible para usuarios no técnicos. Sin embargo, esta accesibilidad a menudo viene a costa de una menor eficiencia en el uso de recursos del sistema, como memoria y CPU. Además, debido a su popularidad, Windows es más susceptible a ataques de malware y virus, lo que plantea preocupaciones de seguridad. Por otro lado, Linux es aclamado por su estabilidad y eficiencia, especialmente en entornos de servidores y sistemas embebidos. No obstante, su alto nivel de personalización y la necesidad de conocimientos técnicos avanzados pueden ser una barrera para muchos usuarios.

- Propuesta del Taller

La propuesta del taller se centra en comparar los paradigmas de programación en serie y en paralelo y evaluar el rendimiento de dos sistemas operativos, Windows y Linux. Este enfoque permite explorar cómo diferentes métodos de programación y plataformas de cómputo influyen en la eficiencia y optimización de programas específicos. Al centrarse en la multiplicación de matrices, tanto en su forma clásica como utilizando una matriz transpuesta, se busca identificar las diferencias en rendimiento bajo condiciones controladas y variadas.

- Metodología Propuesta

La metodología empleada en este taller incluye la selección de algoritmos de multiplicación de matrices implementados en C, un lenguaje conocido por su eficiencia y control sobre la gestión de memoria. Estos algoritmos se ejecutarán en dos sistemas operativos, Windows y Linux, para comparar su rendimiento. Además, se utilizarán múltiples hilos para evaluar el impacto de la programación paralela en comparación con la programación en serie. Basándose en la Ley de los Grandes Números, los experimentos se repetirán numerosas veces para asegurar la precisión y consistencia de los resultados. La métrica principal de evaluación será el tiempo de ejecución medido en microsegundos (μs), complementada por el análisis del uso de recursos como memoria y CPU.

- Resultados Dados el Sistema de Cómputo

Los resultados mostraron que tanto Windows como Linux ofrecieron rendimientos comparables en la ejecución de los algoritmos de multiplicación de matrices. No se encontraron diferencias significativas que sugirieran una ventaja clara de un sistema operativo sobre el otro. Sin embargo, la implementación de la programación paralela mediante el uso de hilos mejoró notablemente el rendimiento, especialmente para matrices de mayor tamaño. La multiplicación utilizando la matriz transpuesta mostró una ligera ventaja en términos de tiempo de ejecución comparado con la multiplicación clásica. Además, se

observó que la mejora en el rendimiento fue significativa hasta el uso de 6 hilos, después de lo cual las mejoras fueron marginales.

8. Conclusiones

El Taller de Evaluación de Rendimiento ha permitido realizar un análisis detallado del rendimiento de programas de multiplicación de matrices bajo diferentes sistemas operativos y paradigmas de programación. A lo largo del informe, se ha evaluado el rendimiento utilizando métricas específicas y se han comparado las plataformas de hardware y software, centrándose en los sistemas operativos Windows y Linux.

La comparativa entre Windows y Linux ha mostrado que ambos sistemas ofrecen un rendimiento similar en la ejecución de algoritmos de multiplicación de matrices, sin diferencias significativas que indiquen una clara ventaja de un sistema operativo sobre el otro. Sin embargo, se han destacado algunas características particulares de cada sistema: mientras Windows es apreciado por su facilidad de uso y amplia compatibilidad, Linux es valorado por su estabilidad y eficiencia, especialmente en entornos más técnicos y personalizables.

La implementación de la programación paralela, mediante el uso de hilos, ha demostrado mejorar significativamente el rendimiento en ambos sistemas operativos. Especialmente para matrices de mayor tamaño, el uso de múltiples hilos ha permitido una ejecución más rápida de los algoritmos. No obstante, se observó que a partir de la utilización de 6 hilos, las mejoras en rendimiento fueron marginales, lo que sugiere un límite en la eficiencia adicional que se puede obtener con un mayor número de hilos en el hardware utilizado.

La comparación entre la multiplicación de matrices clásica y la transpuesta indicó que la versión transpuesta tiene una ligera ventaja en términos de tiempo de ejecución, particularmente para matrices más grandes. Esto se debe a una mayor eficiencia en el acceso y almacenamiento de los elementos de la matriz en memoria, lo que optimiza el uso de recursos computacionales.

Referencias

- Linux vs. Windows: soluciones de alojamiento web. (2021, enero 14). IONOS Digital Guide; IONOS.
<https://www.ionos.es/digitalguide/servidores/know-how/linux-vs-windows-el-gran-cuadro-comparativo/>
- Herrera, J. (2023, diciembre 13). Linux vs Windows vs macOS: similitudes, diferencias, y comparativa de los principales sistemas operativos. Guía Hardware.
<https://www.guiahardware.es/linux-vs-windows-vs-macos/>
- Tanenbaum, A. S. (s/f). Sistemas Operativos Modernos (4ta Edición) Andrew S. Tanenbaum.