

Universidad del Quindío

Programa de Ingeniería de Sistemas y Computación

Proyecto del curso: Multiplicación de matrices grandes

Propósito: Implementar y analizar algoritmos que permitan la multiplicación de matrices grandes aplicando algoritmos iterativos, y algoritmos divide y vencerás.

Requerimiento 1: Elija e implemente 10 algoritmos de los que se plantean en la Tabla 1. Cada uno de los algoritmos elegidos se debe implementar en dos lenguajes de programación diferentes (uno compilado y otro interpretado), no puede usar Java. Se debe especificar una arquitectura para esta implementación. Para este punto también investigue y establezca el orden de complejidad de los algoritmos elegidos. Tenga en cuenta que para los siguientes puntos se asume que son 10 algoritmos.

Requerimiento 2: Generar 8 casos de prueba con matrices de tamaño $n * n$ siendo n factor de (2^n) . Algunos algoritmos tienen esta precondition para funcionar correctamente. Para cada posición $[i][j]$, los números deben contener mínimo 6 dígitos. Los casos de prueba deben ser persistentes (no habrá ingreso manual de los datos). Esto se debe realizar en los dos lenguajes de programación seleccionados.

Requerimiento 3: Registrar de manera persistente el tiempo de ejecución de los algoritmos bajo la misma unidad de tiempo y para los diferentes tamaños de las matrices. Se debe hacer para ambas implementaciones.

Tabla 1. Algoritmos de multiplicación de matrices

Algoritmo de multiplicación de matrices		Tamaño matriz1 (n*n) * Tamaño matriz2 (n*n) n n ⇒ (2^n)		Tamaño matriz1 (n*n) * Tamaño matriz2 (n*n) n n ⇒ (2^n)			Tamaño matriz1 (n*n) * Tamaño matriz2 (n*n) n n ⇒ (2^n)	
		Tiempo de ejecución (TE)		Tiempo de ejecución (TE)			Tiempo de ejecución (TE)	
		Caso 1		Caso 2			Caso 8	
		n* n	TE	n* n	TE	...	n* n	TE
1	NaivOnArray							
2	NaivLoopUnrollingTwo							
3	NaivLoopUnrollingFour							
4	WinogradOriginal							
5	WinogradScaled							
6	StrassenNaiv							
7	StrassenWinograd							
8	III.3 Sequential block							
9	III.4 Parallel Block							
10	III.5 Enhanced Parallel Block							
11	IV.3 Sequential block							
12	IV.4 Parallel Block							
13	IV. 5 Enhanced Parallel Block							
14	V.3 Sequential block							
15	V.4 Parallel Block							

Requerimiento 4: Centralizar los resultados de cada uno de los algoritmos y para ambos lenguajes de programación. La especificación de la arquitectura es clave para cumplir con este requerimiento.

Requerimiento 4: Representar (ej. Diagrama de barras) el tiempo de ejecución de todos los algoritmos para cada uno de los casos de pruebas y en ambos lenguajes de programación. Se debe permitir comparar los resultados de las implementaciones para cada tamaño de entrada definido.

Requerimiento 5: Para cada algoritmo se debe hacer una representación gráfica en la cual se explique la manera en la cual se realiza la multiplicación de matrices. Esa representación se debe hacer en diapositivas. En estos ejemplos se pueden apoyar para la representación:

<https://www.geeksforgeeks.org/c-program-multiply-two-matrices/>

<https://www.geeksforgeeks.org/strassens-matrix-multiplication/>

En la sustentación se analizarán elementos relevantes sobre el comportamiento de los algoritmos de acuerdo con los tamaños de entrada. Cada estudiante debe hacer la sustentación del proyecto.