

ACTIVIDAD: CONCURRENCIA

Santiago Chitiva Contreras

Pontificia Universidad Javeriana
Sistemas Operativos
2024



¿Cómo funciona el Algoritmo Clásico de Multiplicación de Matrices?

Dos matrices A y B son multiplicables si el número de columnas de A coincide con el número de filas de B.

$$M_{m \times n} \times M_{n \times p} = M_{m \times p}$$

El elemento c_{ij} de la matriz producto se obtiene multiplicando cada elemento de la fila i de la matriz A por cada elemento de la columna j de la matriz B y sumándolos.

$$\begin{aligned} A \cdot B &= \begin{pmatrix} 2 & 0 & 1 \\ 3 & 0 & 0 \\ 5 & 1 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 0 \end{pmatrix} = \\ &= \begin{pmatrix} 2 \cdot 1 + 0 \cdot 1 + 1 \cdot 1 & 2 \cdot 0 + 0 \cdot 2 + 1 \cdot 1 & 2 \cdot 1 + 0 \cdot 1 + 1 \cdot 0 \\ 3 \cdot 1 + 0 \cdot 1 + 0 \cdot 1 & 3 \cdot 0 + 0 \cdot 2 + 0 \cdot 1 & 3 \cdot 1 + 0 \cdot 1 + 0 \cdot 0 \\ 5 \cdot 1 + 1 \cdot 1 + 1 \cdot 1 & 5 \cdot 0 + 1 \cdot 2 + 1 \cdot 1 & 5 \cdot 1 + 1 \cdot 1 + 1 \cdot 0 \end{pmatrix} = \\ &= \begin{pmatrix} 3 & 1 & 2 \\ 3 & 0 & 3 \\ 7 & 3 & 6 \end{pmatrix} \end{aligned}$$

Implementación del Algoritmo Clásico de Multiplicación de Matrices en C:

1. Primero que todo, el código comienza con la inclusión de las bibliotecas estándar necesarias para la ejecución del programa, que son: `stdio.h` (para la entrada y salida estándar), `stdlib.h` (para el uso de funciones como “malloc” y “atof”), `sys/time.h` (para manejar el tiempo en el sistema) y `pthread.h` (para la programación de hilos en sistemas POSIX).
2. Se define la constante “RESERVA” que representa el tamaño de la reserva de memoria para las matrices. También se define el struct “datosMM” que almacena información sobre la dimensión de las matrices y el número de hilos, y sirve para encapsular los argumentos de entrada necesarios para la función de multiplicación de matrices, lo que facilita el manejo y el pasode datos a través de los hilos.
3. Se declaran variables globales para las matrices “mA”, “mB” y “mC”, así como un arreglo estático “MEM_CHUNK” para la reserva de memoria.
4. Se implemente la función “multiMatrices”, que realiza la multiplicación de matrices. Toma un argumento de tipo “void*” que contiene información sobre la sección de la matriz que el hilo actual debe procesar. Dentro de esta función se itera sobre las partes asignadas a cada hilo y se calcula la multiplicación de matrices.

5. La función ``initMatrices`` inicializa las matrices ``mA``, ``mB`` y ``mC`` con valores predefinidos.
6. La función ``imprMatrices`` se utiliza para imprimir por pantalla las matrices. Si la dimensión de la matriz es menor que 9, se imprimen todos los elementos de la matriz. En caso contrario, se imprime un mensaje indicando que la matriz es demasiado grande para ser mostrada completamente.
7. En la función “main”, se verifica que se proporcionen los argumentos adecuados desde la línea de comandos (la dimensión de las matrices y el número de hilos). Luego se asigna memoria para las matrices, se inicializan, se imprimen, se crean los hilos para la multiplicación de matrices, se espera a que todos los hilos terminen su ejecución y finalmente se imprime la matriz resultante.