

PONTIFICIA UNIVERSIDAD CATÓLICA MADRE Y MAESTRA
FACULTAD DE CIENCIAS E INGENIERÍA
ESCUELA DE INGENIERÍA EN COMPUTACIÓN Y TELECOMUNICACIONES

PROGRAMACIÓN PARALELA Y CONCURRENTE

Practica #1

Multiplicación de matrices con MPI en Java

Objetivo:

Implementar un programa en Java utilizando la biblioteca MPI-Java que realice la multiplicación paralela de dos matrices, centrándose en los conceptos de arquitectura de memoria compartida, arquitectura de memoria distribuida y paso de mensajes.

Instrucciones:

1. Arquitectura de Memoria Compartida:
 - a. Inicialice las matrices A y B solo en el proceso raíz (rango 0).
2. Arquitectura de Memoria Distribuida:
 - a. Distribuya las filas de las matrices A y B a todos los procesos utilizando la función Bcast() de MPI.
3. Paso de Mensajes:
 - a. Cada proceso calculará una porción de la matriz resultante C utilizando el algoritmo de multiplicación de matrices.
 - b. Utilice mensajes para comunicar las porciones de las matrices A, B y C entre los procesos.
4. Recopilación de Resultados:
 - a. Reúna los resultados parciales de cada proceso en el proceso raíz utilizando la función Gather() de MPI.
5. Pruebas y Verificación:
 - a. Verifique la corrección del programa probándolo con matrices de diferentes tamaños.

- b. Compare los resultados con la multiplicación de matrices secuencial para garantizar la precisión de la implementación paralela.

Entregables:

- Código fuente del programa en Java utilizando la biblioteca MPI-Java. El código debe subirse a un repositorio público en GitHub.
- Documentación en formato README que incluya una breve descripción del programa, instrucciones para compilar y ejecutar el programa, ejemplos de entrada y salida, y análisis del rendimiento en función del número de procesos utilizados. La documentación también debe subirse al repositorio en GitHub.
- El enlace al repositorio público en GitHub debe ser proporcionado en la plataforma de entrega de la tarea. Asegúrate de que el repositorio sea público para poder acceder al proyecto.

Nota: Se requiere la instalación y configuración de la biblioteca MPI-Java en el entorno de desarrollo. Se sugiere utilizar herramientas como Eclipse con el complemento MPJ Express o IntelliJ IDEA con soporte para MPI-Java para facilitar el desarrollo y la depuración del programa.