

Práctica Nro. 4

Programación con MPI

Se seleccionará un ejercicio que los alumnos deberán entregar en grupos de dos personas.

Pautas:

Compilar en Linux Openmpi:

mpicc -o salidaEjecutable archivoFuente

Ejecutar en Openmpi:

En una sola maquina:

mpirun -np cantidadDeProcesos ejecutable

En un cluster de máquinas:

mpirun -np cantidadDeProcesos -machinefile archivoMaquinas ejecutable

El formato de archivo de máquinas es:

maquina1 slot=cantidad de procesadores de la maquina1

maquina2 slot=cantidad de procesadores de la maquina2

...

maquinaN slot=cantidad de procesadores de la maquinaN

1. Resolver una multiplicación de matrices de $N \times N$ y analizar los tiempos de comunicación utilizando:

Solo operaciones de envío y recepción de mensajes simples.

Broadcast y operaciones colectivas.

Utilizar tamaños de matrices de 512, 1024 y 2048. Probar con 4 y 8 procesos.

2. Realizar un algoritmo paralelo que dada una matriz A de $N \times N$ obtenga el valor máximo, el valor mínimo y valor promedio de A , luego debe armar una matriz B de la siguiente forma:

- Si el elemento $a_{i,j} < \text{promedio}(A)$ entonces $b_{i,j} = \min(A)$.
- Si el elemento $a_{i,j} > \text{promedio}(A)$ entonces $b_{i,j} = \max(A)$.
- Si el elemento $a_{i,j} = \text{promedio}(A)$ entonces $b_{i,j} = \text{promedio}(A)$.

Probar con 4 y 8 procesos.

3. Realizar un algoritmo paralelo que ordene un vector de N elementos por mezcla. Paralelizar con 4 y 8 procesos.
4. Dado un texto representado en una matriz T de $N \times N$, se debe realizar un algoritmo paralelo que obtenga la lista de palabras de T , y luego determine la cantidad de veces que aparece cada palabra en el texto quedándose sólo con las cinco palabras más frecuentes. Probar con 4 y 8 procesos.