



Sistemas Paralelos - 2019

Trabajo Práctico N° 4 *Paradigmas de programación paralela*



Objetivo de la práctica:

Comprender los diferentes paradigmas de programación paralela, y analizar las ventajas y desventajas de aplicarlos en distintos problemas.

Bibliografía:

- I. “High Performance Cluster Computing: Programming and Application” - Vol 2. Rajkumar Buyya. Prentice Hall. 1999. <http://www.buyya.com/cluster/v2chap1.pdf>
- II. Characteristics of Workloads Using the Pipeline Programming Model. Christian Bienia and Kai Li. Computer Architecture. Springer Berlin Heidelberg. pp 161-171. isbn=978-3-642-24322-6. 2012. <https://parsec.cs.princeton.edu/publications/bienia10pipeline.ps>

Ejercicios

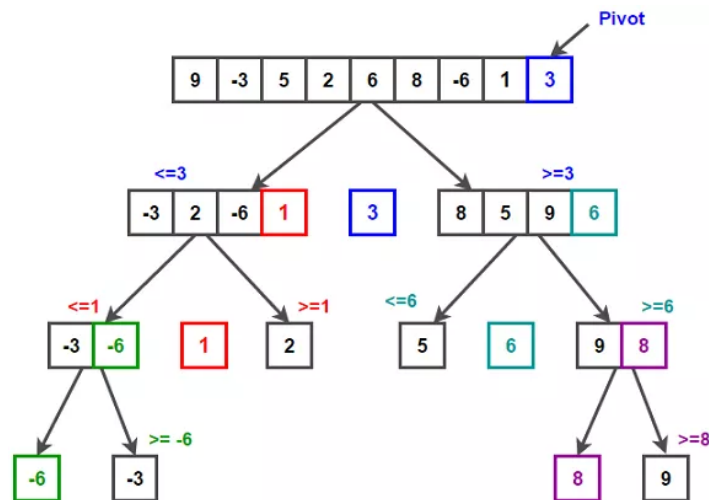
1) Mencione dos motivaciones para utilizar el paradigma de programación paralela denominado data pipelining (ver bibliografía II).

2) Se tiene un arreglo de dos dimensiones, y una función que es evaluada para cada elemento del arreglo. El cómputo de cada elemento es independiente de los demás. El problema es computacionalmente intensivo. El programa serie calcula un elemento por vez en orden secuencial. Se espera una solución paralela que escale tanto en número de procesadores como en tamaño del problema.

¿Es posible de paralelizar el problema? Si su respuesta es afirmativa, determinar el paradigma de programación paralela más conveniente para resolver el problema. Detallar: cómo particionaría el problema, el trabajo que realizaría cada tarea, las comunicaciones entre tareas, y si la distribución de trabajo a las tareas será estática o dinámica.

```
do j = 1,n
  do i = 1,n
    a(i,j) = fcn(i,j)
  end do
end do
```

3) Queremos implementar el ordenamiento de un arreglo de números enteros por medio del algoritmo quicksort. La solución más simple que se le ocurre, ¿a qué paradigma corresponde? Explicar cómo se haría el particionado y distribución de trabajo entre las tareas.



4) Suponga un simulador del tránsito de vehículos en una ciudad. El asfalto está congelado en las esquinas de las calles, y dos vehículos circulan a alta velocidad por calles perpendiculares. Si ambos conductores mantienen la velocidad, es altamente probable que choquen entre ellos, en la intersección de ambas calles. Si chocan, se requiere un cómputo intensivo para determinar cuál sería la nueva configuración de los semáforos que permitan desviar el tránsito por calles alternativas. Suponiendo que, antes de saber si los vehículos chocan o no, hay procesadores ociosos a causa de dependencias de datos. ¿Qué paradigma cree que sería útil en este caso?

- 5) Se quiere resolver el juego de la vida utilizando una máquina de memoria distribuida.
- Describa dos soluciones, una utilizando el paradigma master-worker y otra utilizando el paradigma SPMD.
 - ¿Cuál de las dos opciones tiene un volumen de comunicaciones menor? Justificar con un ejemplo.
 - Considerando la solución SPMD, determinar (a partir de las dependencias de datos) qué cómputo es posible realizar mientras se realizan las comunicaciones.