

## Actividad 3: Diseño de algoritmo paralelo.

Cómputo de Alto Rendimiento

**Luis Fernando Izquierdo Berdugo**

### Pseudocódigo

Crear un array principal con números del 1 al 100.

Dividir el array principal en 10 arrays con la misma cantidad de datos.

Pasar un array a cada uno de los procesadores

En cada procesador:

- Eliminar los múltiplos de 2 menores o iguales a 100 del array

- Eliminar los múltiplos de 3 menores o iguales a 100 y mayores de 3 del array

- Eliminar los múltiplos de 5 menores o iguales a 100 y mayores de 5 del array

- Eliminar los múltiplos de 7 menores o iguales a 100 y mayores de 7 del array

- Devolver el array resultante al proceso principal

Crear un array con los arrays resultantes de cada procesador.

### Etapas

Partición:

Para esto se dividirá el conjunto principal en 10 conjuntos de 10 números.

Comunicación:

Debido a que cada subconjunto trabajará con datos diferentes, no será necesaria la comunicación, ya que no se podría correr el riesgo de eliminar el número dos veces del conjunto principal.

Agrupación:

Solamente habrá una comunicación entre todos los procesadores al momento de asignarles su conjunto de número y al momento de devolver los números primos. Debido a que todos tienen la misma cantidad de datos en su conjunto, se tiene el mismo costo. Se podría subdividir para particionar la obtención de los múltiplos de 2, 3, 5 y 7, pero esto podría aumentar el costo.

Asignación:

Tendremos cuatro procesadores y cada uno de ellos hará el mismo proceso con su subconjunto de datos. Todos obtendrán los múltiplos de 2, 3, 5 y 7 menores o iguales a 100 de su subconjunto asignado, eliminándolos del conjunto que se les pasó inicialmente.

### Justificación del algoritmo

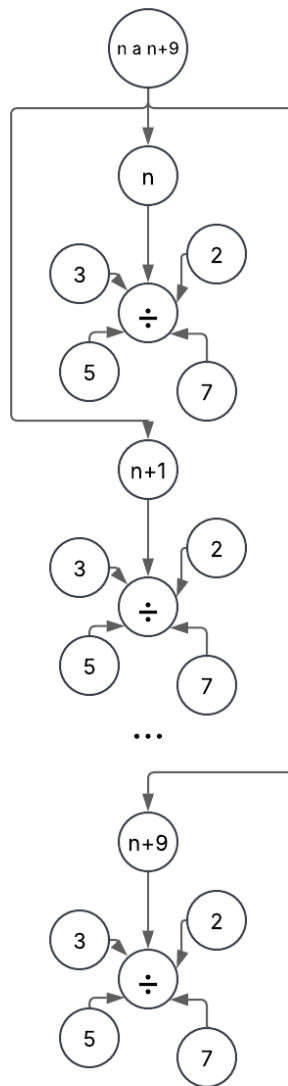
Se eligió dividir el algoritmo de esta manera debido a que se disminuye la comunicación entre procesadores en comparación con la otra opción ideada, que era dividir los procesos de obtención de múltiplos de 2, 3, 5 y 7.

Este algoritmo tiene ventaja con conjuntos pequeños como el presentado, ya que al hacerlo más grande se tendrían que aumentar muchísimo los procesadores y eso terminaría siendo más costoso que otras opciones. Una desventaja es esperar el tiempo de procesamiento para cada múltiplo (esperar para 2, luego para 3, etc.) sin embargo, esto asegura que todos los conjuntos tienen el mismo costos de comunicación y cómputo.

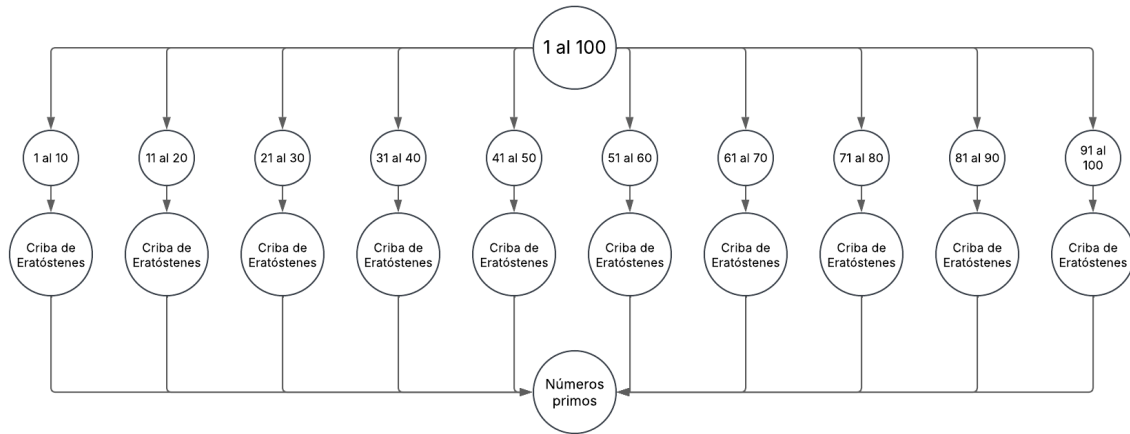
## **DAG**

Debido a la complejidad del algoritmo de criba de eratóstenes, se crearon dos DAG, uno que describe el proceso dentro de la criba y el otro que muestra el algoritmo paralelizado.

# Criba de Eratóstenes



Algoritmo Criba de Eratóstenes  
para un conjunto del 1 al 100



**Referencias:**

Arellano Vázquez, M. (s.f.). *Diseño de algoritmos paralelos*. Ciudad de México, México: INFOTEC.

Arellano Vázquez, M. (s.f.). *Introducción*. Ciudad de México, México: INFOTEC.