

# Análisis

Iván Alba Gómez

## 1. `primosParalelos.java`

En este algoritmo podemos observar cómo se utiliza la ecuación de subramanian. Se obtiene el número de procesadores lógicos del procesador mediante la instrucción: `Runtime.getRuntime().availableProcessors()` y se supone que el coeficiente de bloqueo es 0, puesto que no hay posibilidad de que se produzca bloqueo en este algoritmo.

En este caso, vemos que el número de hilos es igual al número de procesadores lógicos que disponemos, ya que, como el coeficiente de bloqueo es 0, al hacer la siguiente ecuación:  $\text{procesadoresLogicos} / 1 = \text{procesadoresLogicos}$ .

Acto seguido, se divide el rango entre el número de hilos obtenido en la ecuación de Subramanian, con lo cual, cada hilo trabajará en un fragmento de dicho rango, para así obtener el resultado final más rápidamente.

A continuación, podremos ver una tabla donde se comparan los tiempos de ejecución de los algoritmos de búsqueda de números primos, tanto en su versión secuencial como en la versión paralela, para valores distintos de rango:

Rango	Secuencial	Paralelo
1.000	0	0
10.000	0	0
100.000	0	0
1.000.000	0	0
10.000.000	19	9
20.000.000	50	21

Como podemos observar, a medida que vamos aumentando el rango, la diferencia de tiempos entre las versiones secuencial y paralela también aumenta.