

# Teoría de Colas

Clara Téllez

18 de febrero de 2020

## 1. Objetivo

La práctica consiste en examinar las diferencias en los tiempos de ejecución de los diferentes ordenamientos cuando se varía el número de los núcleos asignados al cluster [1].

## 2. Metodología

Para la simulación se usó una matriz de datos de números primos descargado de <https://primes.utm.edu/lists/small/millions/> que fue modificada para tener una matriz resultante con los números primos desde uno hasta cinco millones con un total de 348,513 entradas. A partir de esta matriz se creó un vector de datos para realizar la simulación. Teniendo el vector de datos de números primos, se creó un vector de números no primos mediante la multiplicación del vector de números primos por cinco.

Para examinar los tiempos de ejecución con diferente número de núcleos asignados al cluster y hacer el respectivo tratamiento estadístico se usó R en su versión 3.6.2.

La práctica consistió en ejecutar la búsqueda de números primos en los vectores creados, asignando las tareas a realizar a cuatro y siete núcleos. Las tareas se realizaron en diferente orden, es decir, primero realizó la búsqueda en los números primos y después en los números no primos, después realizó la búsqueda primero en los números no primos y después en los números primos y por último, realizó la búsqueda tomando los números de forma aleatoria.

Se realizaron diez réplicas para cada orden y se evaluó el tiempo que tardó cada ejecución.

## 3. Resultados y Discusión

El resumen estadístico de los tiempos de ejecución está dado en el cuadro 1

Los datos muestran claramente el aumento en el tiempo de ejecución al usar un menor número de núcleos en la ejecución de tareas. Al analizar los tiempos

Cuadro 1: datos estadísticos de los tiempos de ejecución (en segundos) para 4 y 7 núcleos y en su diferente orden. Original: Primero la matriz de números y primos y después la matriz de números no primos; Invertida: primero la matriz de números no primos y después la matriz de números primos; Aleatoria: ejecución aleatoria de las matrices. min: mínimo, 1st qu: primer cuartil, med: mediana, mean: media, 3rs qu: tercer cuartil, max: máximo

| Ordenamiento          | min   | 1st qu | median | mean   | 3rd qu | max    |
|-----------------------|-------|--------|--------|--------|--------|--------|
| Original (4 núcleos)  | 471.6 | 577.5  | 666.5  | 1792.9 | 1706.7 | 7503.3 |
| Invertido (4 núcleos) | 577.6 | 577.8  | 627.9  | 1246.6 | 758.8  | 5284.9 |
| Aleatorio (4 núcleos) | 587.6 | 590.1  | 592.3  | 1623.2 | 893.5  | 7215.1 |
| Original (7 núcleos)  | 319.1 | 320.4  | 321.5  | 350.2  | 328.5  | 510.2  |
| Invertido (7 núcleos) | 321.3 | 322.3  | 322.8  | 412.9  | 388.3  | 873.5  |
| Aleatorio (7 núcleos) | 323.1 | 326.0  | 326.7  | 374.6  | 327.8  | 579.4  |

de ejecución según el ordenamiento en las tareas se observa una incongruencia entre los datos, mientras que para los análisis realizados con cuatro núcleos, fue el ordenamiento invertido el que se ejecutó de manera más rápida, fue éste ordenamiento, el más tardado en la ejecución con siete núcleos. Adicionalmente, se calculó el tiempo empleado para la ejecución de los tres ordenamientos en ambos núcleos. En el análisis realizado por cuatro núcleos el tiempo total fue de aproximadamente diez horas (35904.89), mientras que trabajando con siete núcleos se tardó aproximadamente tres horas (11408.63). Estos datos confirman la eficiencia que ofrece el aprovechamiento de los núcleos al calendarizar las tareas a realizar.

## 4. Conclusiones

El número de núcleos asignados a la ejecución de tareas tiene un gran impacto en la rapidez con la que éstas se realizan, sin embargo, no fue posible establecer la influencia en el ordenamiento de las tareas sobre el tiempo de ejecución.

## Referencias

- [1] E. Schaeffer. *Práctica 3: Teoría de Colas*. Feb. de 2020. URL: <https://elisa.dyndns-web.com/teaching/comp/par/p3.html>.