**Diferencias en hilos, con los métodos “start()” y “run()”**

Cuando ejecutamos el programa con el método “start()”, los datos de los 3 hilos aparecen mezclados, es decir aparecen intervalos de cada uno de los 3 hilos, lo que significa que se ejecutan en paralelo. Por el contrario, si ejecutamos el programa con el método “run()”, se imprimen los valores de cada uno de los hilos ordenados, hasta que el hilo 1 no acabe de imprimir, no inicia el segundo y así mismo con el 3 hilo, es decir en forma secuencial.

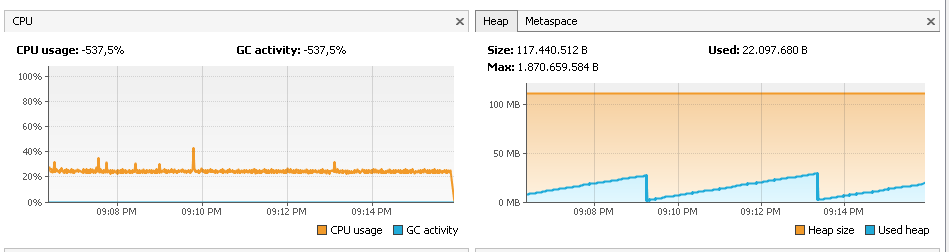
Esto nos demuestra que el método “start()” ejecuta los hilos de forma paralela sin necesidad de esperar a que cada uno de los hilos termine su procedimiento, por el contrario el método “run()” ejecuta los hilos de forma secuencial.

**Parte III Evaluación de Desempeño**

**Longitud: 100.000**

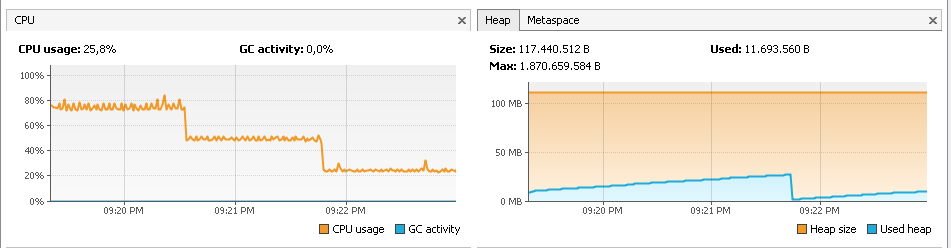
1. Un solo hilo.

**Tiempo:** 9 Minutos 31 Segundos



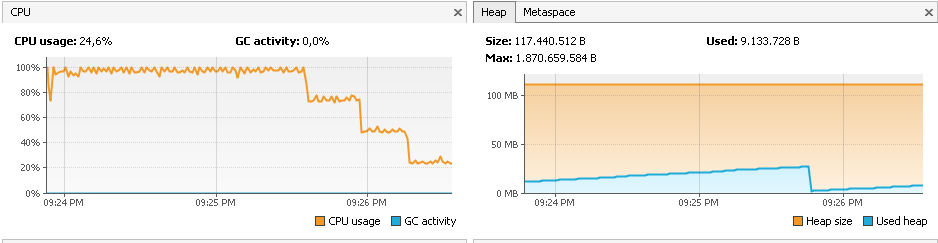
1. Tantos hilos como núcleos de procesamiento

**Tiempo:** 4 Minutos 14 Segundos



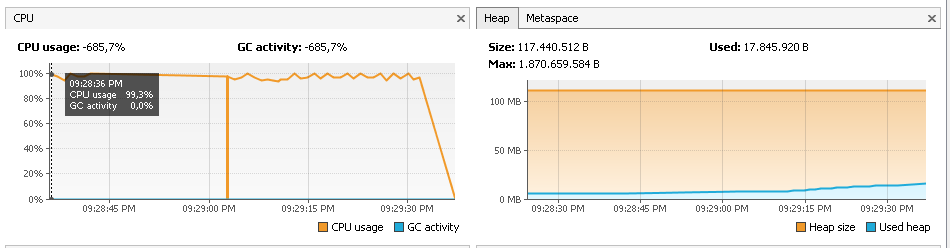
1. Tantos hilos como el doble de núcleos de procesamiento.

**Tiempo:** 3 Minutos 04 Segundos



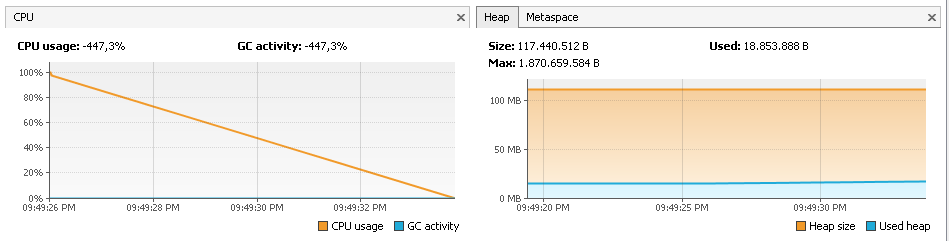
1. 200 hilos.

**Tiempo:** 2 Minutos 39 Segundos

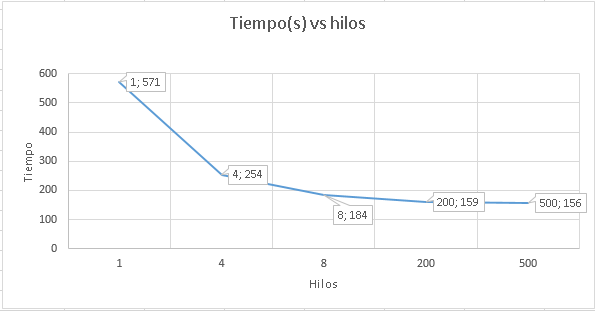


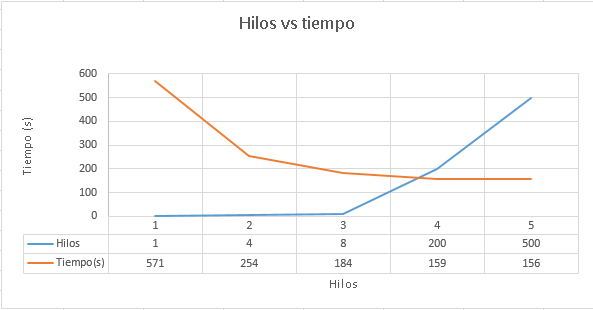
1. 500 hilos

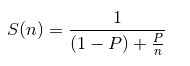
**Tiempo:** 2 Minutos 36 segundos









1. **Según la** [**ley de Amdahls**](https://www.pugetsystems.com/labs/articles/Estimating-CPU-Performance-using-Amdahls-Law-619/#WhatisAmdahlsLaw?)**: [](https://github.com/ARSW-ECI/Parallelism-JavaThreads-Introduction/blob/master/img/ahmdahls.png), donde S(n) es el mejoramiento teórico del desempeño, P la fracción paralelizable del algoritmo, y n el número de hilos, a mayor n, mayor debería ser dicha mejora. Por qué el mejor desempeño no se logra con los 500 hilos?, cómo se compara este desempeño cuando se usan 200?.**

No necesariamente entre mayor sea la cantidad de hilos el desempeño será mejor. Cada computador tiene una cantidad determinada de núcleos que permite manejar distintos procesos y estos sus respectivos hilos. La cantidad de hilos que se pueden trabajar en paralelo depende de la cantidad de núcleos, por eso es que el rendimiento no depende solo de la cantidad de hilos, en este caso aunque el desempeño si fue mejor con 500 hilos por 3 segundos, la prueba se volvió a realizar con 700 hilos y volvió a dar el mimo resultado que si se hubiera realizado con 500. Como vimos la CPU en los últimos casos ya estaba cercana al 100% lo que quiere decir que todos los hilos posibles estaban trabajando al tiempo, a otros les tocaba esperar a que otros hilos terminaran para poder iniciar (forma secuencial). Por eso no necesariamente más hilos optimizan un programa.

1. **Cómo se comporta la solución usando tantos hilos de procesamiento como núcleos comparado con el resultado de usar el doble de éste?.**
2. **De acuerdo con lo anterior, si para este problema en lugar de 500 hilos en una sola CPU se pudiera usar 1 hilo en cada una de 500 máquinas hipotéticas, la ley de Amdahls se aplicaría mejor?. Si en lugar de esto se usaran c hilos en 500/c máquinas distribuidas (siendo c es el número de núcleos de dichas máquinas), se mejoraría?. Explique su respuesta.**