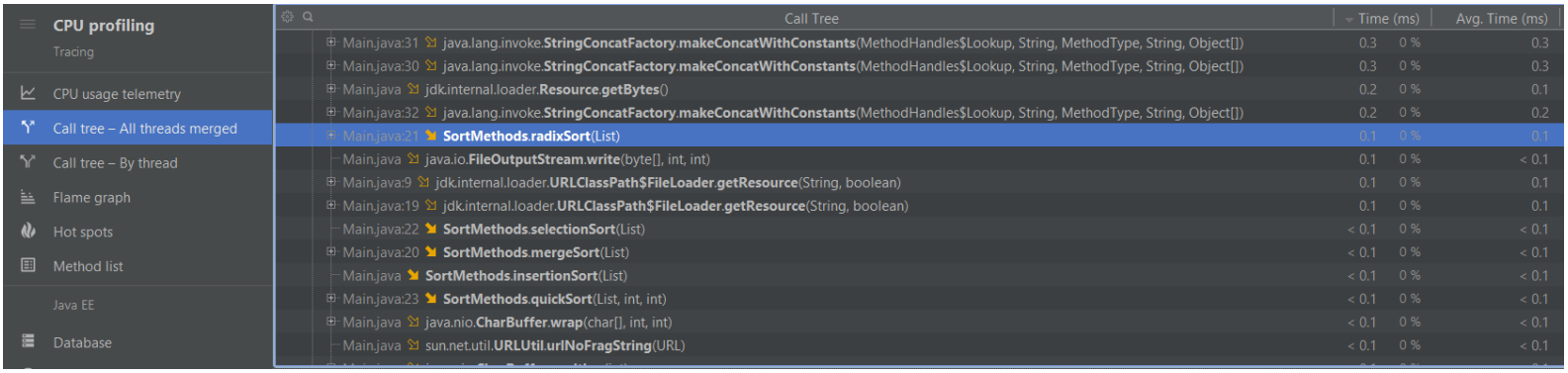
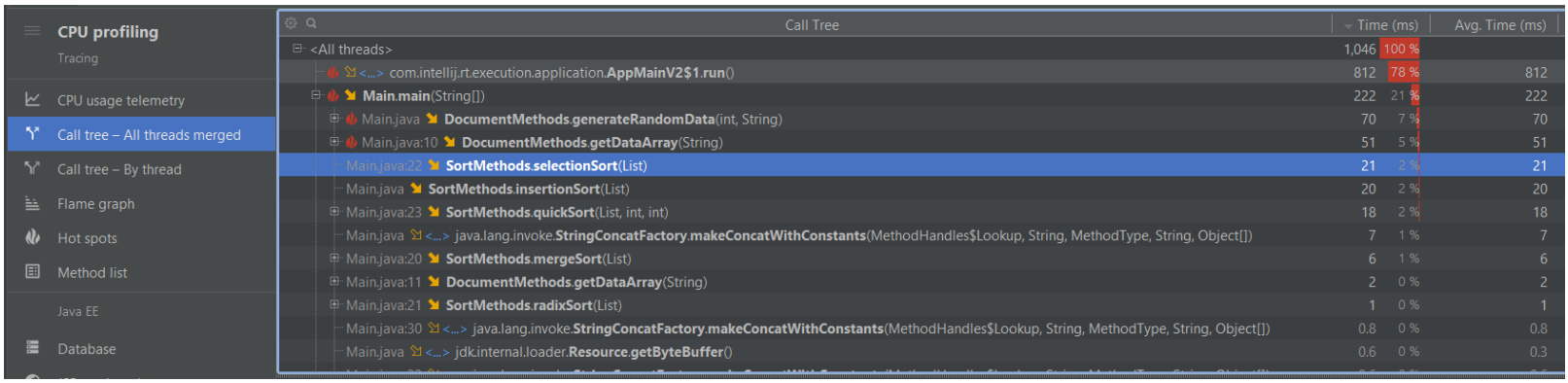
Pablo Ruiz 18259

Tiempo en correr los sorts pruebas de Junit y análisis del profiler

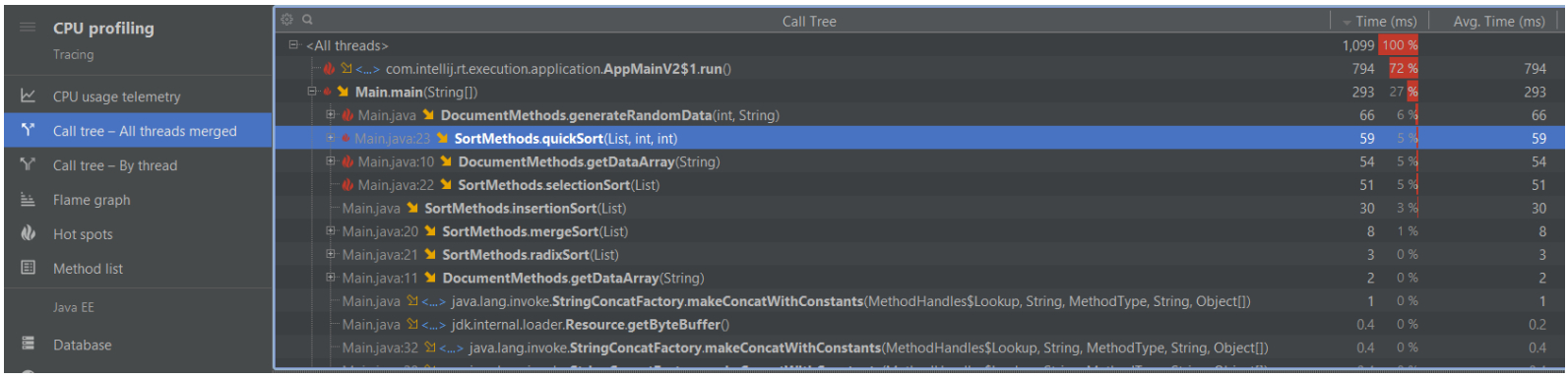
10 Elementos:



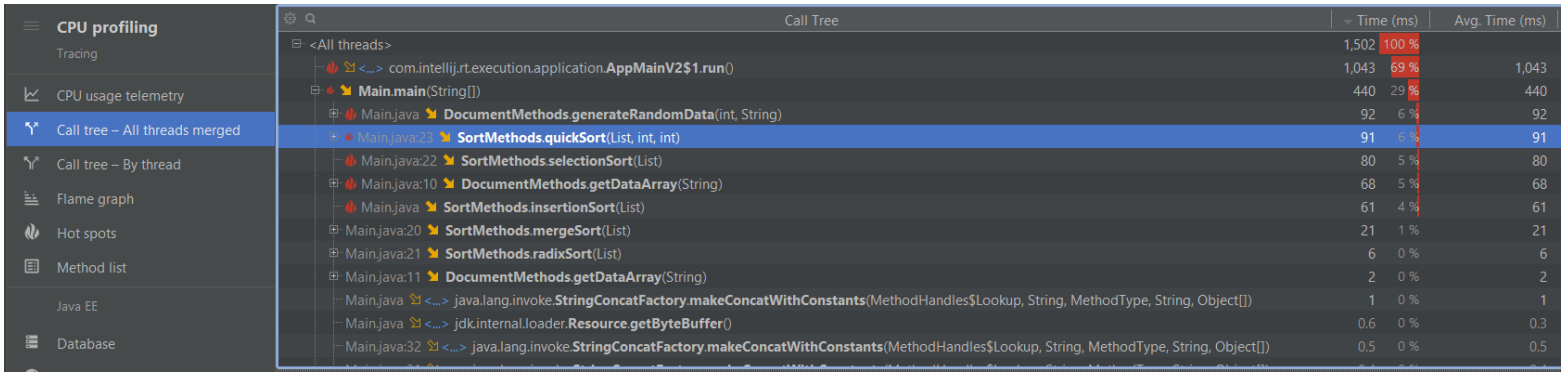
375 Elementos:



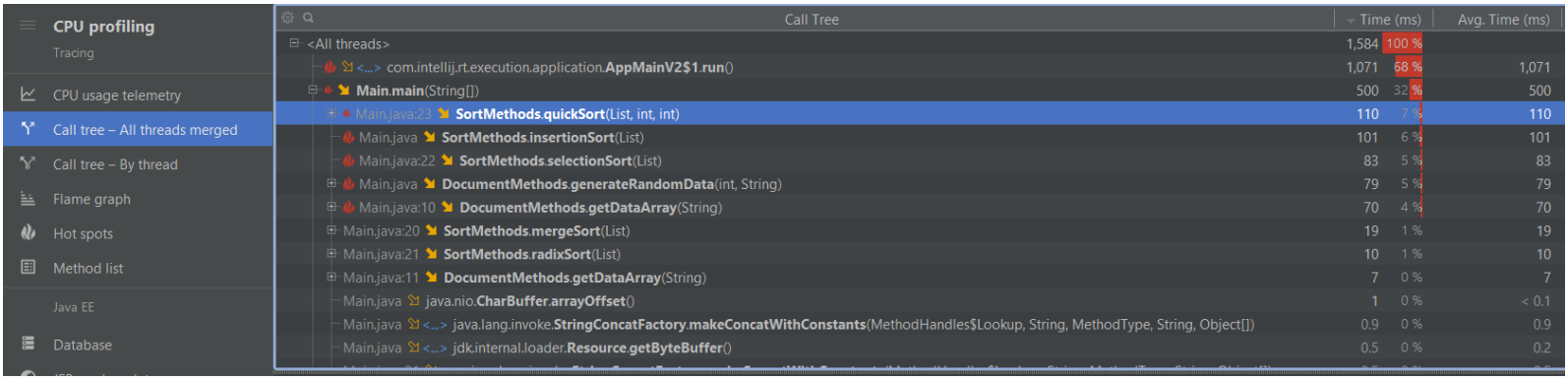
750 Elementos:



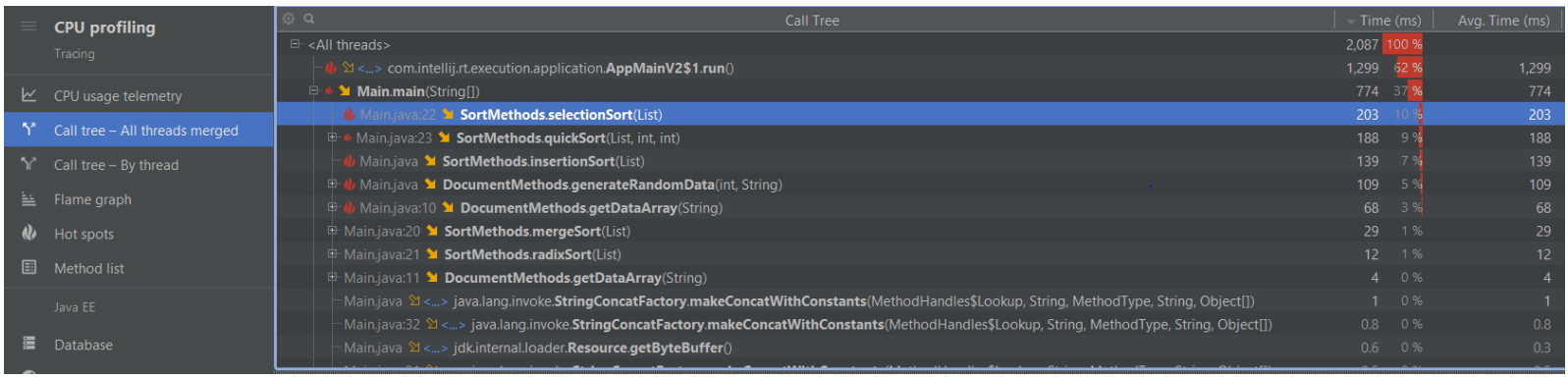
1125 Elementos:



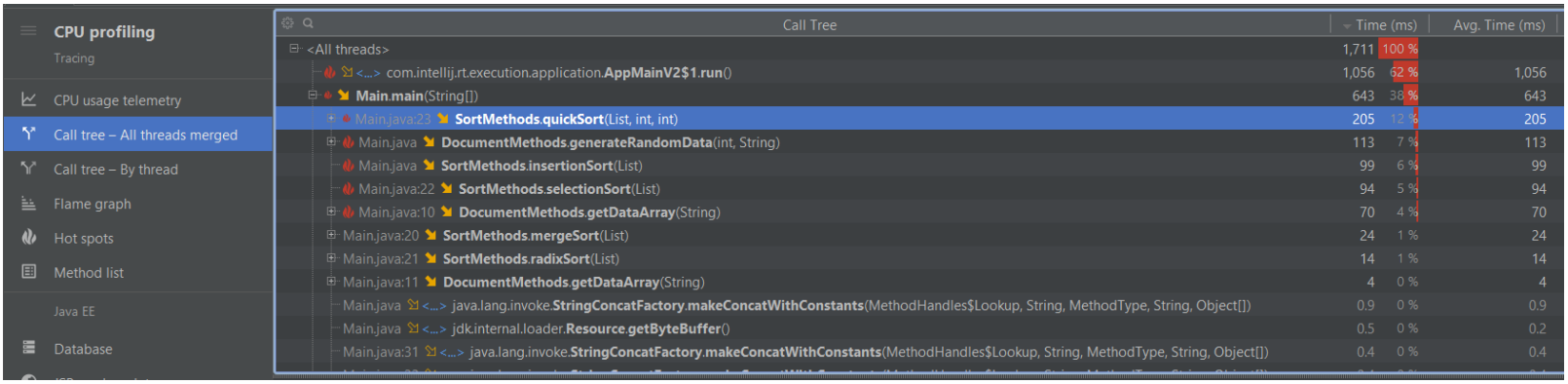
1500 Elementos:



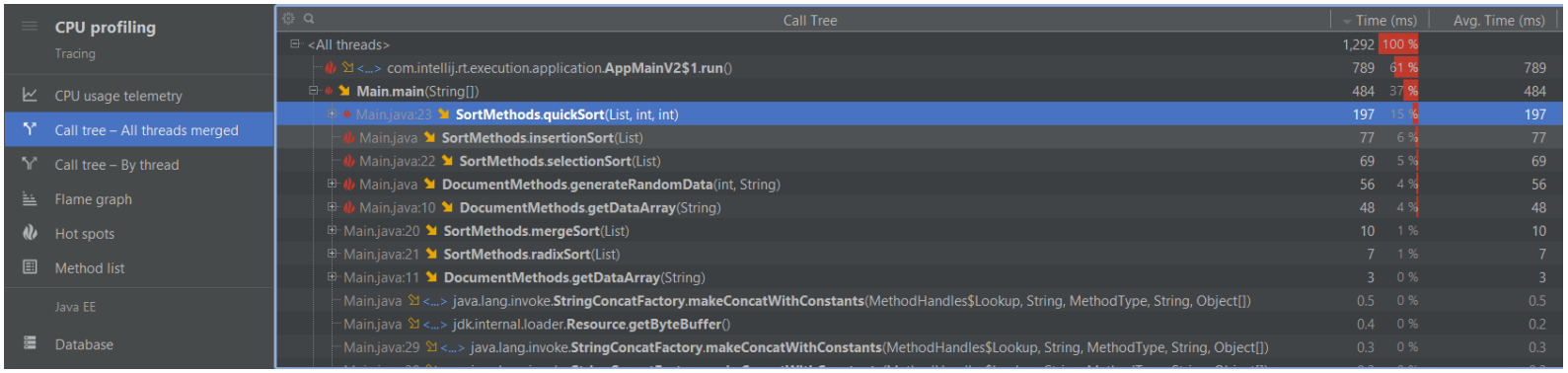
1875 Elementos:



2250 Elementos:



2625 Elementos:



3000 Elementos: Muestra los tiempos de ejecución de los sorts en un conjunto de datos desordenados (blanco) y en el mismo conjunto ordenado (verde).

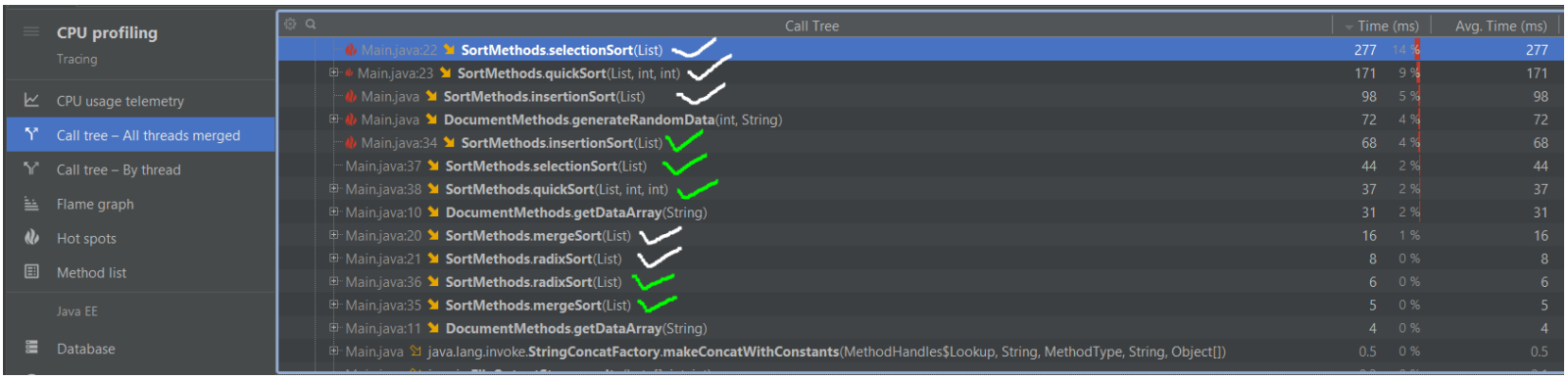


Tabla 1. Tiempo de corrida de sorts según el número de datos

	Tiempo de corrida en millisegundos				
Número de datos	Merge	Quick	Insertion	Radix	Selection
10	0.1	0.1	0.1	0.1	0.1
375	6	18	20	1	21
750	8	59	30	3	51
1125	21	91	61	6	80
1500	19	110	101	10	83
1875	29	188	139	12	203
2250	24	205	99	14	94
2625	10	197	77	7	69
3000	16	171	98	8	277

Figura 1. Tiempo de corrida de sorts

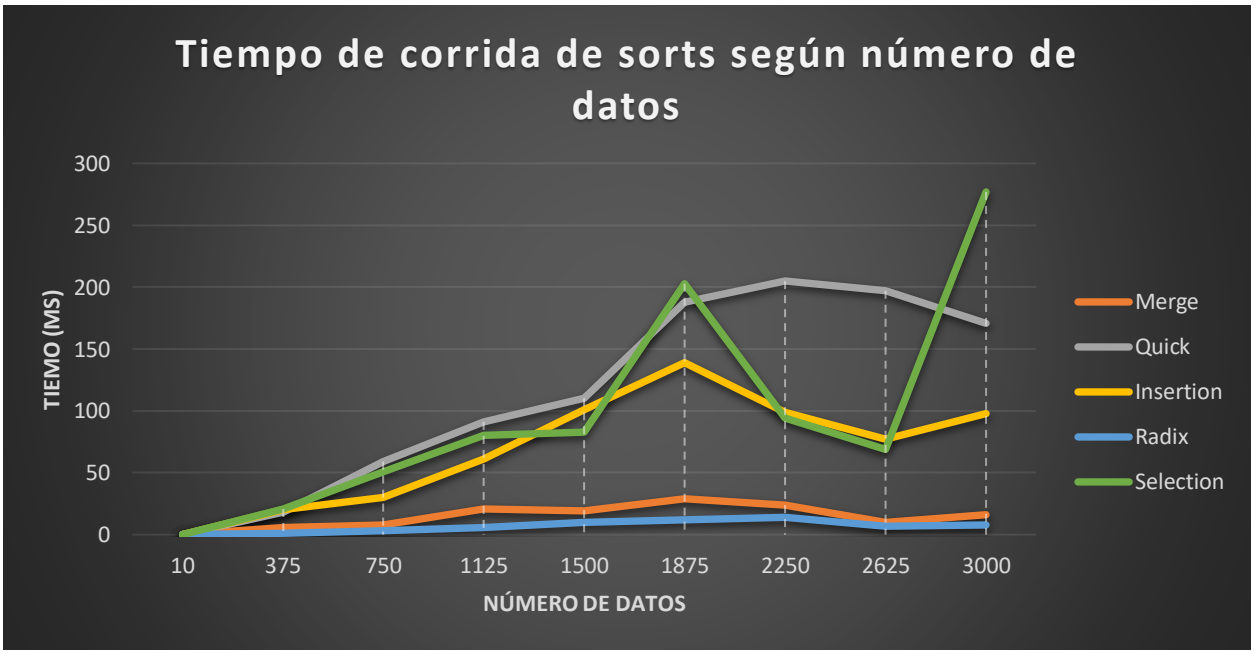
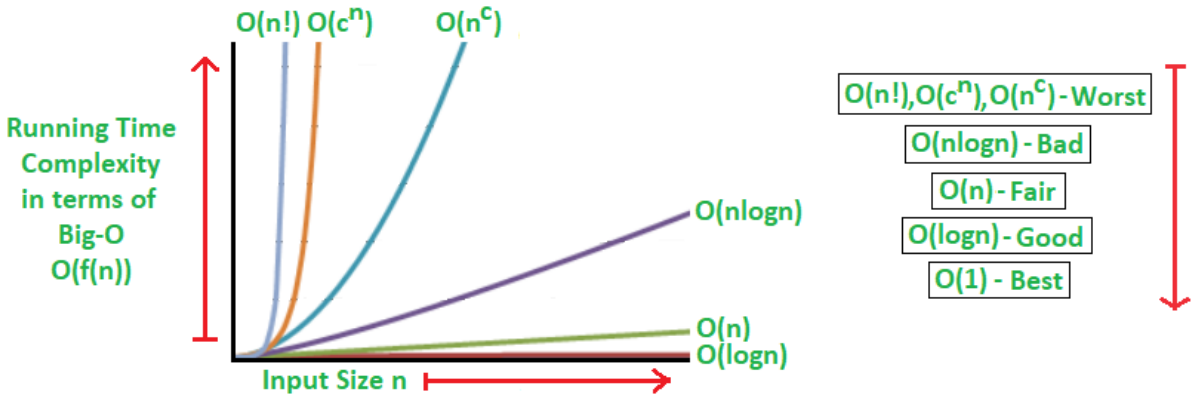


Figura 2. Tiempo de corrida teórico de sorts



Tomado de GeeksForGeeks https://www.geeksforgeeks.org/analysis-algorithms-big-o-analysis/

En este caso, el merge sort y Quicksort deberían de tener una forma de O(nlogn), el Selection e Insertion de O(n ^ 2) y el radix de O(n).

Análisis del profiler

El profiler utilizado fue Yourkit. Se empleó a través de la descarga de un plugin para IntelliJ, en donde se permite realizar un profiling del código directamente desde e IDE. Para ello, se descargó Yourkit desde su sitio web (https://www.yourkit.com/) y se configuró en la máquina. Luego, desde la aplicación instalada, se configuró el plugin para IntelliJ. Finalmente, desde IntelliJ se configuró para que este hiciera Tracing del CPU. Se utilizó a través de IntelliJ, en donde solamente se le daba click al botón de profiling y se abría el Profiler. Al abrirse, se abría el snapshot creado y se selecciona los métodos en la pestaña izquierda del programa. Al abrir esto, se podían visualizar los tiempos de corrida.

Figura 3. Muestra de pruebas aprobadas de los sorts en Junit