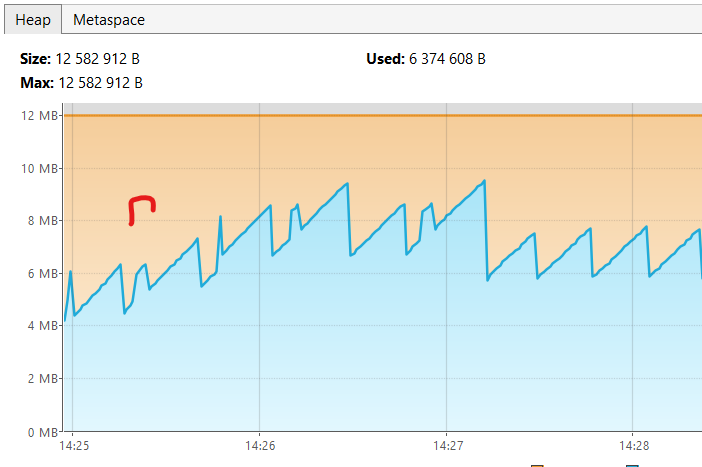
-XX:+UseParallelGC 

Начальное состояние памяти:



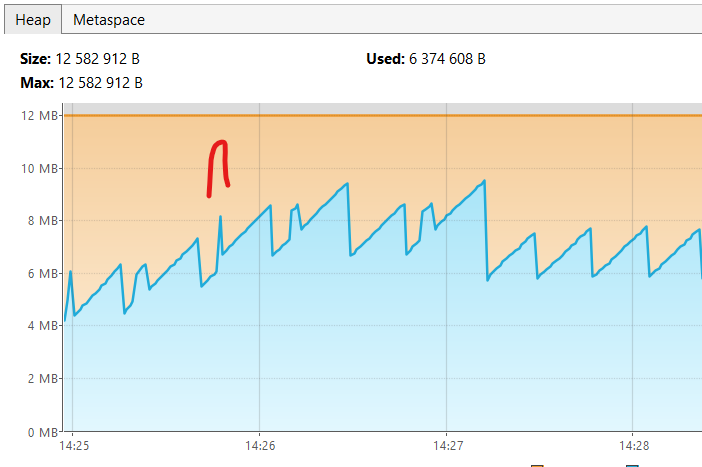
создаем массив на 250000 элементов.



Рост массивов int[] на 1млн байт



Сортировка слиянием:



MergeSort Начало сортировки - 14:06:05.983548900

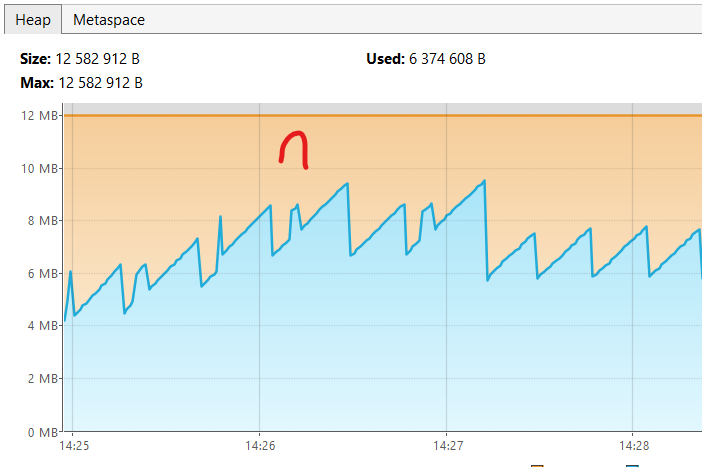
MergeSort Конец сортировки - 14:06:06.041656900



Сортировка происходит за 0,06сек

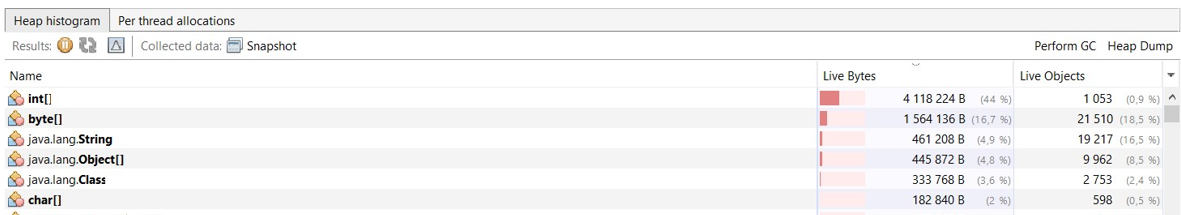
Значительный рост массивов int[] из-за создания копии массива при сортировке

Сортировка методом вставки:



InsertSort Начало сортировки - 14:06:10.474878

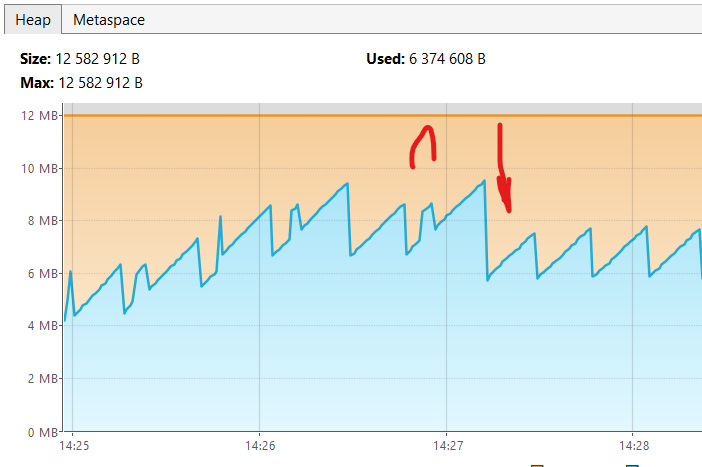
InsertSort Конец сортировки - 14:06:15.066643200



Сортировка происходит за 5сек

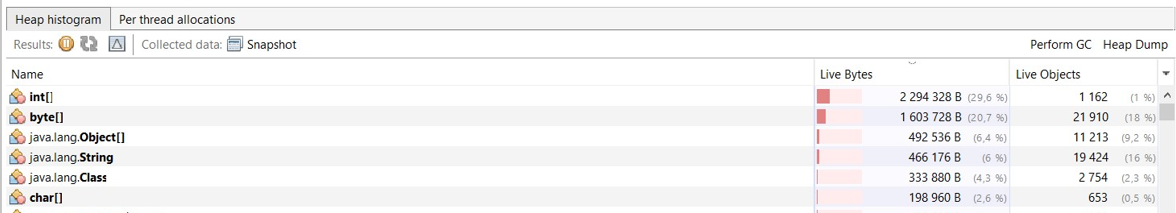
Рост массивов int[] на 1млн байт (размер нашего массива)

Сортировка пузырьком:



BubbleSort Начало сортировки - 14:06:18.469136300

BubbleSort Конец сортировки - 14:07:46.199481700



Сортировка происходит за 88сек

Рост массивов int[] на 1млн байт (размер нашего массива) + произошла полная сборка

[0.005s][info][gc] Using Parallel

[1.060s][info][gc] GC(0) Pause Young (Allocation Failure) 3M->1M(11M) 1.682ms

[2.589s][info][gc] GC(1) Pause Young (Allocation Failure) 4M->2M(11M) 1.716ms

[2.632s][info][gc] GC(2) Pause Young (Allocation Failure) 5M->3M(11M) 1.123ms

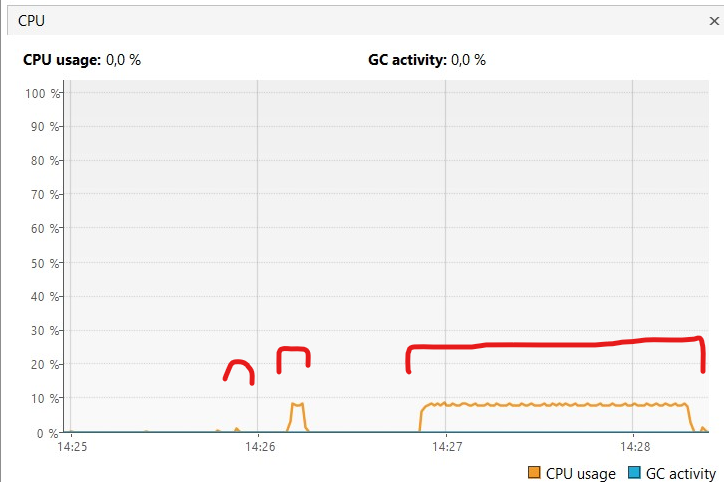
[2.663s][info][gc] GC(3) Pause Young (Allocation Failure) 6M->4M(11M) 1.749ms

[2.680s][info][gc] GC(4) Pause Young (Allocation Failure) 7M->5M(11M) 2.007ms

[2.915s][info][gc] GC(5) Pause Young (Allocation Failure) 8M->6M(10M) 2.200ms

[3.020s][info][gc] GC(6) Pause Young (Allocation Failure) 8M->6M(11M) 2.079ms

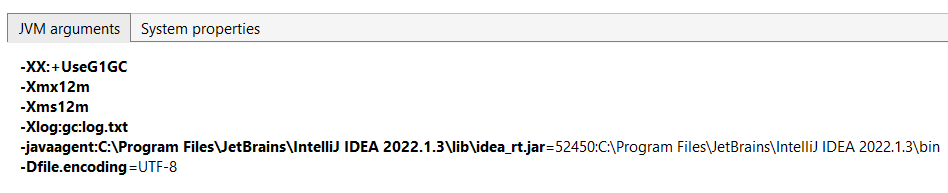
[3.029s][info][gc] GC(7) Pause Full (Ergonomics) 6M->4M(11M) 9.319ms

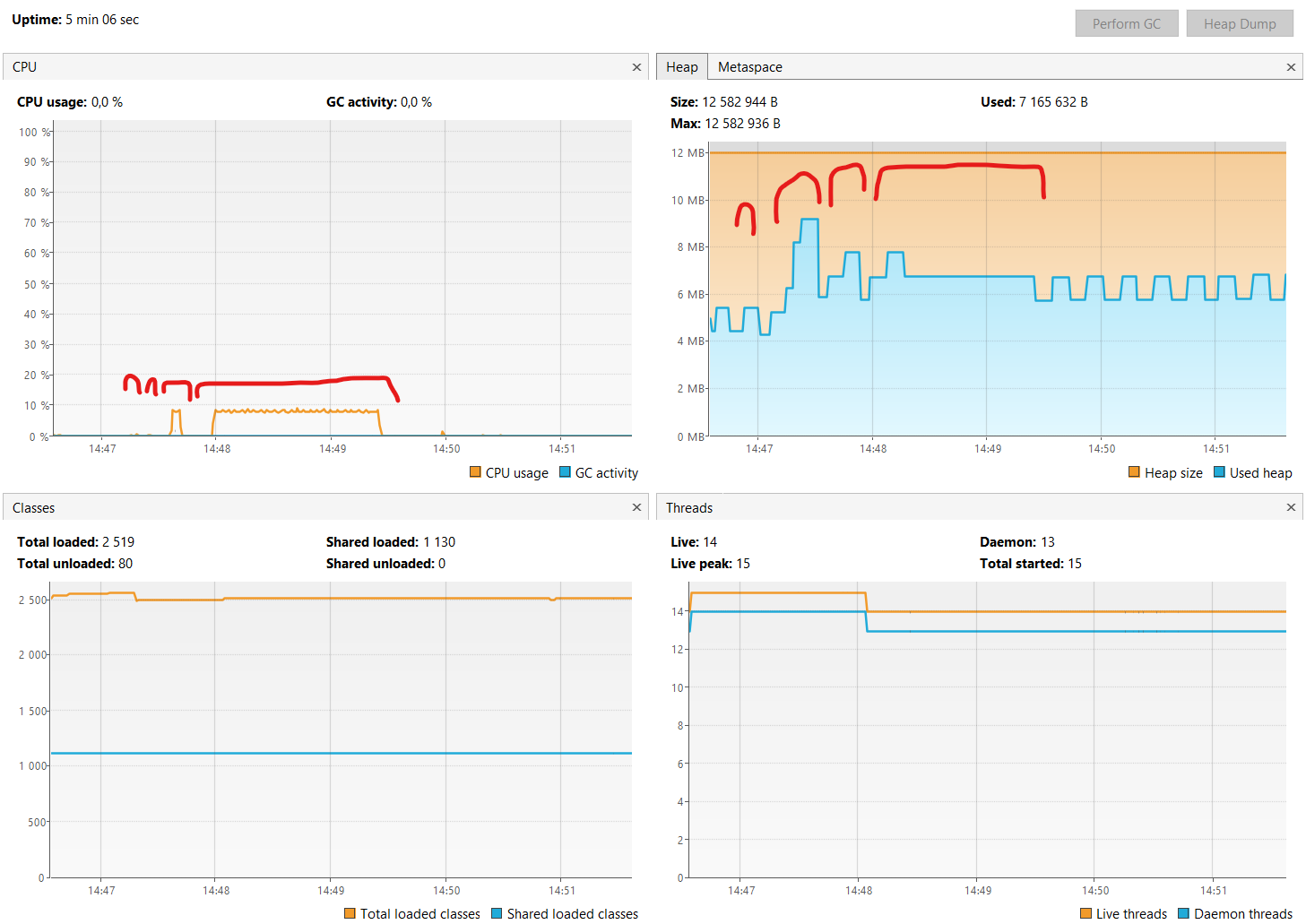


Загрузка процессора по типам сортировки:

слиянием / методов вставки / пузырьком

UseG1GC





[0.006s][info][gc] Using G1

[3.041s][info][gc] GC(0) Pause Young (Normal) (G1 Evacuation Pause) 4M->2M(12M) 2.289ms

[3.065s][info][gc] GC(1) Pause Young (Normal) (G1 Evacuation Pause) 3M->2M(12M) 1.986ms

[3.078s][info][gc] GC(2) Pause Young (Normal) (G1 Evacuation Pause) 3M->2M(12M) 0.729ms

[3.105s][info][gc] GC(3) Pause Young (Normal) (G1 Evacuation Pause) 4M->2M(12M) 1.130ms

[3.138s][info][gc] GC(4) Pause Young (Normal) (G1 Evacuation Pause) 4M->3M(12M) 1.420ms

[3.162s][info][gc] GC(5) Pause Young (Normal) (G1 Evacuation Pause) 6M->3M(12M) 1.117ms

[3.285s][info][gc] GC(6) Pause Young (Normal) (G1 Evacuation Pause) 6M->3M(12M) 1.013ms

[3.402s][info][gc] GC(7) Pause Young (Normal) (G1 Evacuation Pause) 6M->4M(12M) 1.994ms

[4.350s][info][gc] GC(8) Pause Young (Normal) (G1 Evacuation Pause) 6M->4M(12M) 2.226ms

[13.452s][info][gc] GC(9) Pause Young (Normal) (G1 Evacuation Pause) 6M->4M(12M) 4.313ms

[28.464s][info][gc] GC(10) Pause Young (Normal) (G1 Evacuation Pause) 6M->4M(12M) 1.182ms

[35.169s][info][gc] GC(11) Pause Young (Concurrent Start) (G1 Humongous Allocation) 5M->4M(12M) 1.279ms

[35.169s][info][gc] GC(12) Concurrent Undo Cycle

[35.169s][info][gc] GC(12) Concurrent Undo Cycle 0.088ms

[47.192s][info][gc] GC(13) Pause Young (Concurrent Start) (G1 Humongous Allocation) 6M->5M(12M) 1.298ms

[47.192s][info][gc] GC(14) Concurrent Mark Cycle

[47.197s][info][gc] GC(14) Pause Remark 7M->7M(12M) 2.314ms

[47.198s][info][gc] GC(14) Pause Cleanup 8M->8M(12M) 0.018ms

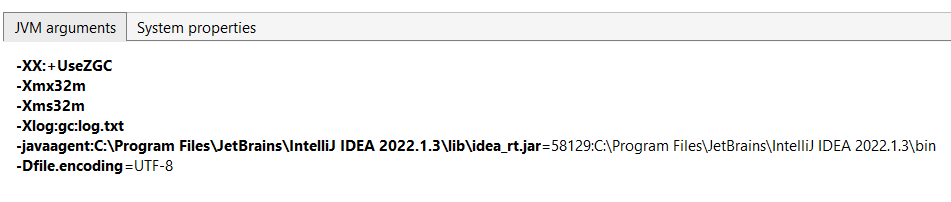
[47.198s][info][gc] GC(14) Concurrent Mark Cycle 6.241ms

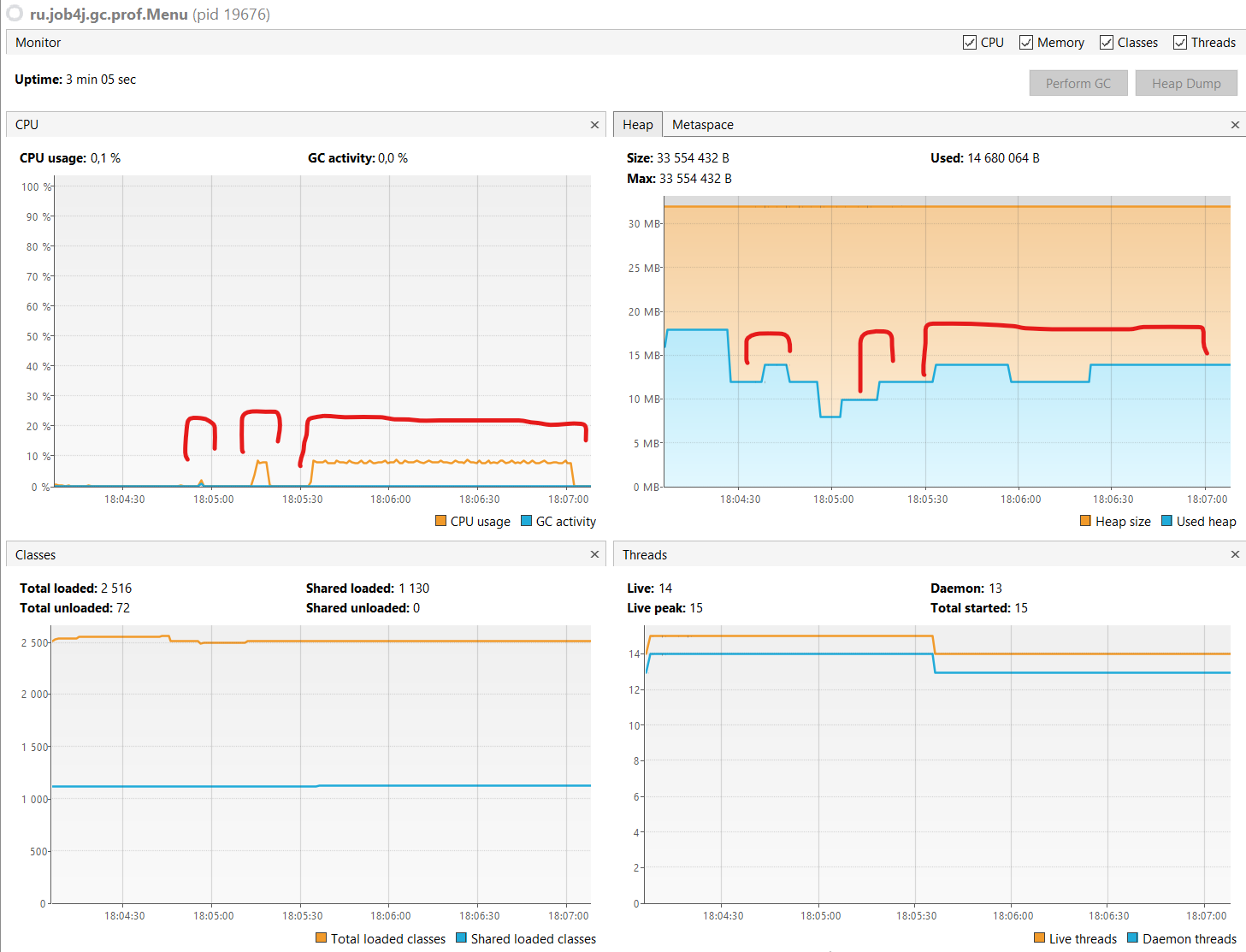
[47.200s][info][gc] GC(15) Pause Young (Normal) (G1 Evacuation Pause) 8M->6M(12M) 1.134ms

[47.205s][info][gc] GC(16) Pause Young (Concurrent Start) (G1 Evacuation Pause) 7M->6M(12M) 0.957ms

[47.205s][info][gc] GC(17) Concurrent Mark Cycle

UseZGC





[0.008s][info][gc] Using The Z Garbage Collector

[0.143s][info][gc] GC(0) Garbage Collection (Warmup) 4M(12%)->4M(12%)

[3.136s][info][gc] GC(1) Garbage Collection (Warmup) 12M(38%)->8M(25%)

[3.245s][info][gc] GC(2) Garbage Collection (Warmup) 16M(50%)->6M(19%)

[4.248s][info][gc] GC(3) Garbage Collection (Proactive) 16M(50%)->10M(31%)

[23.536s][info][gc] GC(4) Garbage Collection (Proactive) 14M(44%)->8M(25%)

[34.536s][info][gc] GC(5) Garbage Collection (Proactive) 12M(38%)->10M(31%)

[42.642s][info][gc] GC(6) Garbage Collection (Proactive) 14M(44%)->8M(25%)

[52.642s][info][gc] Allocation Stall (main) 2.578ms

[52.643s][info][gc] GC(7) Garbage Collection (Allocation Rate) 22M(69%)->22M(69%)

[52.738s][info][gc] GC(8) Garbage Collection (Allocation Stall) 32M(100%)->8M(25%)

[52.866s][info][gc] GC(9) Garbage Collection (Allocation Rate) 8M(25%)->8M(25%)

[52.951s][info][gc] GC(10) Garbage Collection (Allocation Rate) 8M(25%)->8M(25%)

[53.051s][info][gc] GC(11) Garbage Collection (Allocation Rate) 8M(25%)->8M(25%)

[53.139s][info][gc] GC(12) Garbage Collection (Allocation Rate) 8M(25%)->8M(25%)

[53.253s][info][gc] GC(13) Garbage Collection (Allocation Rate) 8M(25%)->8M(25%)

[53.341s][info][gc] GC(14) Garbage Collection (Allocation Rate) 8M(25%)->8M(25%)

[59.538s][info][gc] GC(15) Garbage Collection (Proactive) 12M(38%)->8M(25%)

[70.647s][info][gc] GC(16) Garbage Collection (Proactive) 12M(38%)->8M(25%)

[71.938s][info][gc] GC(17) Garbage Collection (Proactive) 12M(38%)->12M(38%)

[90.443s][info][gc] GC(18) Garbage Collection (Proactive) 16M(50%)->12M(38%)

[113.645s][info][gc] GC(19) Garbage Collection (Proactive) 16M(50%)->12M(38%)

[139.649s][info][gc] GC(20) Garbage Collection (Proactive) 16M(50%)->12M(38%)

[166.646s][info][gc] GC(21) Garbage Collection (Proactive) 16M(50%)->12M(38%)