Project 1: Comparison-based Sorting Algorithms

ITCS 6114 – Algorithms and Data Structures

By Ria Banerjee & TJ Bah

**Implement the following Sorting Algorithms:**

|  |  |  |
| --- | --- | --- |
| Algorithm | Time Complexity | File Name |
| Insertion Sort | O(n2) | insertionSort.zip |
| Merge Sort | O(n log n) | mergeSort.zip |
| Heapsort | O(n log n) | heapSort.zip |
| In-Place Quicksort | O(n log n) | inPlaceQuickSort.zip |
| Modified Quicksort | O(n log n) | modifiedQuickSort.zip |

**Notes and Observations:**

**Insertion Sort:**

**Generally has the worst time complexity but not in all cases. In the special case that the input array is already sorted, the insertion sort is the top performer. This may have implications on nearly sorted arrays as well.**

**Merge Sort:**

**Heapsort:**

**In-Place Quicksort:**

**Modified Quicksort:**

**Small Sample Size Execution Results Graphs (<= 5 K):**

Small size Random Order Input Graph Analysis:

For input sizes less than or equal to 5000, that are in random order our average worst performer is insertion sort that runs in O(n^2) time. The next worst to best performers are in-place quicksort, merge sort and heap sort. These three sorting algorithms run in O(n log n) time. Our results seem to reflect this notion.

Insertion Sort:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Input Size n | Run # 1 | Run # 2 | Run #3 | Average |
| 1 K | 5990700 | 5611000 | 4312600 | 5304766.667 |
| 2 K | 9473300 | 9359600 | 9164600 | 9332500 |
| 4 K | 17934000 | 14865800 | 19392000 | 17397266.67 |
| 5 K | 17435000 | 18201900 | 26204800 | 20613900 |

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

Merge Sort:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Input Size n | Run # 1 | Run # 2 | Run #3 | Average |
| 1 K | 1715800 | 1617000 | 1053100 | 1461966.667 |
| 2 K | 2481700 | 2069600 | 2579900 | 2377066.667 |
| 4 K | 3234600 | 2759600 | 3379000 | 3124400 |
| 5 K | 3444001 | 3125200 | 3832901 | 3467367.333 |

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

In-Place Quick Sort:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Input Size n | Run # 1 | Run # 2 | Run #3 | Average |
| 1 K | 2064900 | 2914800 | 2278700 | 2419466.667 |
| z K | 3836700 | 3926000 | 2972300 | 3578333.333 |
| 4 K | 5667300 | 5690300 | 5446900 | 5601500 |
| 5 K | 7141100 | 8121000 | 7311800 | 7524633.333 |

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

Small size Already Sorted Input Graph Analysis:

In the special case that the input array is already sorted, our worst performer is the in-place quicksort, and our best performer is insertion sort. One interesting observation is that the in-place quicksort outperforms the merge sort initially when n is less than or equal to 1000. However, for n sizes greater than or equal to 2000 the execution time for in place quick sort grows dramatically. Heap and merge sort follow a similar growth rate with the edge going to heap sort.

Insertion Sort ( Special Case 1: Already Sorted Input ):

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Input Size n | Run # 1 | Run # 2 | Run #3 | Average |
| 1 K | 29500 | 49600 | 66500 | 48533.33333 |
| 2 K | 101300 | 100300 | 109000 | 103533.3333 |
| 4 K | 179900 | 228300 | 220300 | 209500 |
| 5 K | 154000 | 293300 | 208800 | 218700 |

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

Merge Sort ( Special Case 1: Already Sorted Input ):

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Input Size n | Run # 1 | Run # 2 | Run #3 | Average |
| 1 K | 923300 | 999600 | 1133200 | 1018700 |
| 2 K | 1390800 | 1839300 | 1530300 | 1586800 |
| 4 K | 2451500 | 1912100 | 2236400 | 2200000 |
| 5 K | 2354800 | 2106700 | 2550100 | 2337200 |

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

In-Place Quick Sort ( Special Case 1: Already Sorted Input ):

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Input Size n | Run # 1 | Run # 2 | Run #3 | Average |
| 1 K | 1024600 | 622800 | 972700 | 873366.6667 |
| 2 K | 1667000 | 1581300 | 1669100 | 1639133.333 |
| 4 K | 4423500 | 3949000 | 3824100 | 4065533.333 |
| 5 K | 5166200 | 7626100 | 7337200 | 6709833.333 |

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

Small size Reversely Sorted Input Graph Analysis:

Our worst performer in the special case that the input is reversely sorted is insertion sort. This is the worst case for the algorithm. Similarly one of the scenarios that cause the in-place quicksort to perform at its worst-case running time is when the input array is reversely sorted. This produces an O(n^2) runtime that is on par with the insertion sort in this case. Our best performers in this case are the merge and heap sort. These two algorithms both have a runtime of O(n log n).

Insertion Sort ( Special Case 2: Reversely Sorted Input ):

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Input Size n | Run # 1 | Run # 2 | Run #3 | Average |
| 1 K | 2548900 | 3325800 | 2996800 | 2957166.667 |
| 2 K | 4119100 | 4521300 | 3875400 | 4171933.333 |
| 4 K | 16944100 | 11688400 | 15790000 | 14807500 |
| 5 K | 28739500 | 16734900 | 26779800 | 24084733.33 |

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

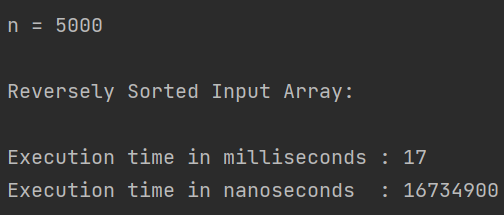
Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

Text

Description automatically generated  Text

Description automatically generated

Merge Sort ( Special Case 2: Reversely Sorted Input ):

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Input Size n | Run # 1 | Run # 2 | Run #3 | Average |
| 1 K | 219800 | 255400 | 246100 | 240433.3333 |
| 2 K | 454700 | 329100 | 459600 | 414466.6667 |
| 4 K | 1140700 | 1164400 | 1021800 | 1108966.667 |
| 5 K | 1337300 | 918100 | 1317600 | 1191000 |

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

In-Place Quick Sort ( Special Case 2: Reversely Sorted Input ):

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Input Size n | Run # 1 | Run # 2 | Run #3 | Average |
| 1 K | 527000 | 736700 | 539700 | 601133.3333 |
| 2 K | 995500 | 1562300 | 1519300 | 1359033.333 |
| 4 K | 4517600 | 4706200 | 3762300 | 4328700 |
| 5 K | 8268900 | 7417000 | 10251700 | 8645866.667 |

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

Text

Description automatically generated Text

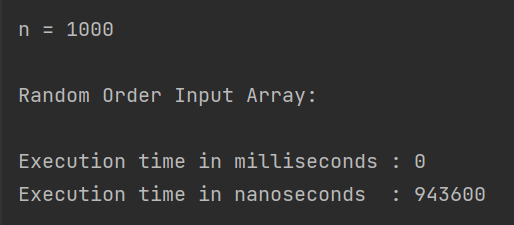
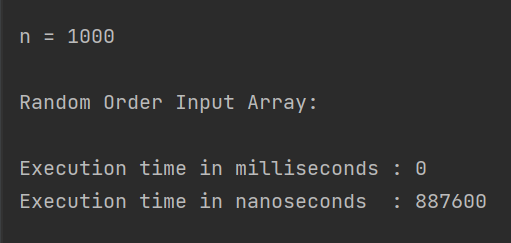
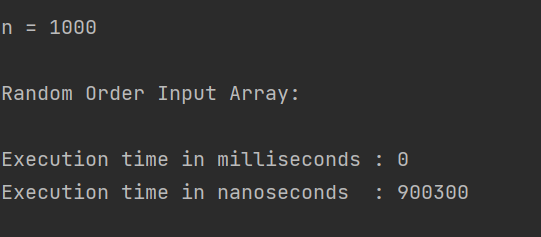
Description automatically generated Text

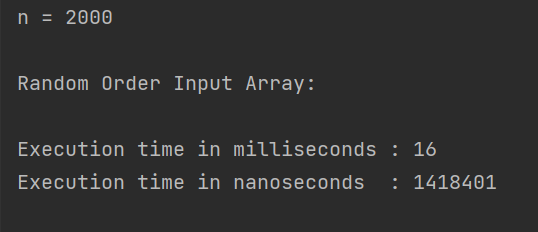
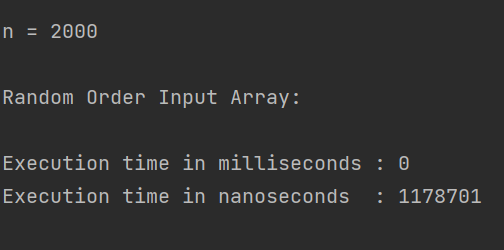
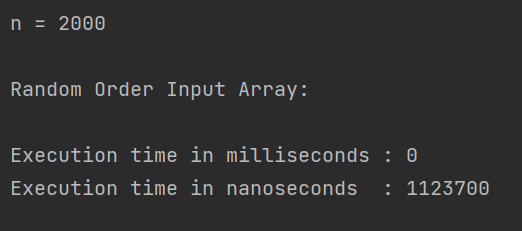
Description automatically generated

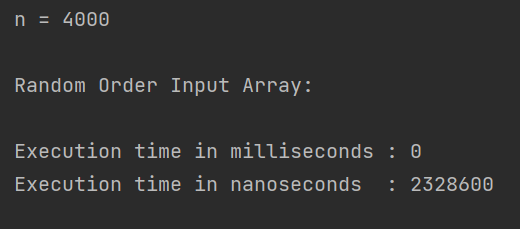
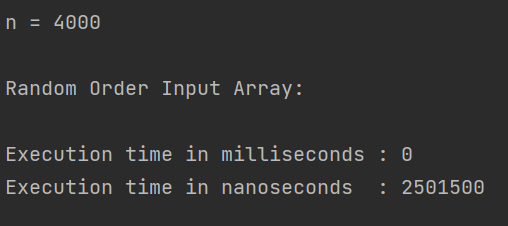
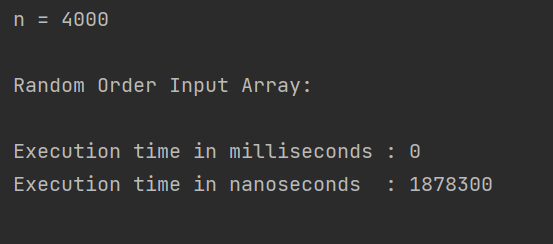
Heap sort:

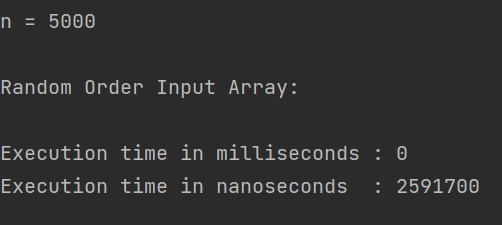
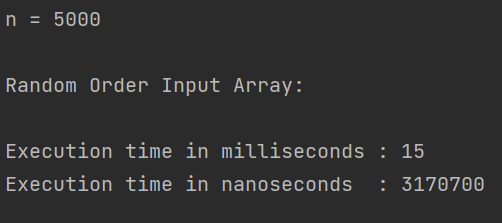
Table: Input size, and time in nanoseconds

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Input Size | Run #1 | Run #2 | Run #3 | Average |
| 1K | 900300 | 887600 | 943600 | 910500 |
| 2K | 1123700 | 1178701 | 1418401 | 1240267 |
| 4K | 1878300 | 2501500 | 2328600 | 2236133 |
| 5K | 3170700 | 2477000 | 2591700 | 2746466 |





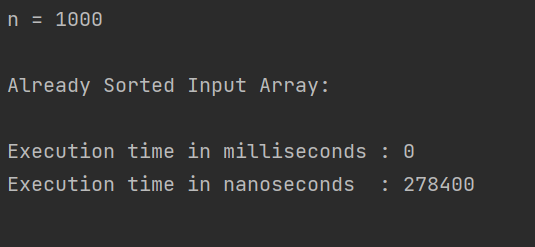
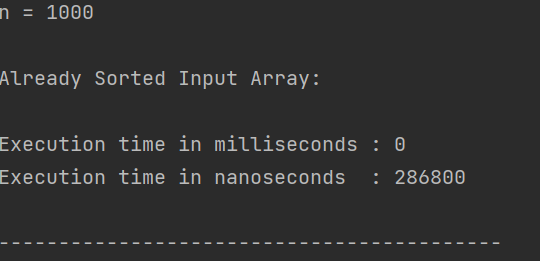


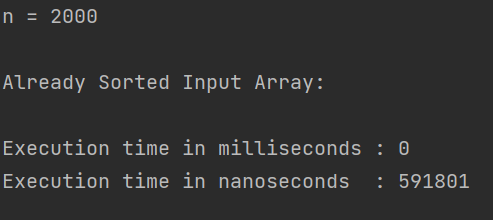
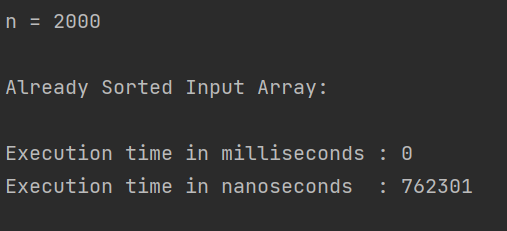
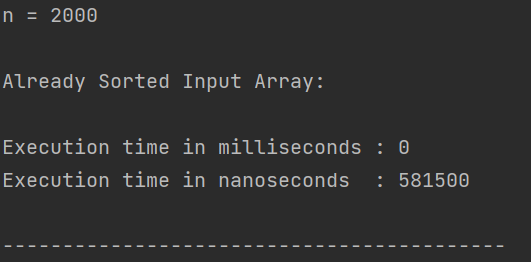


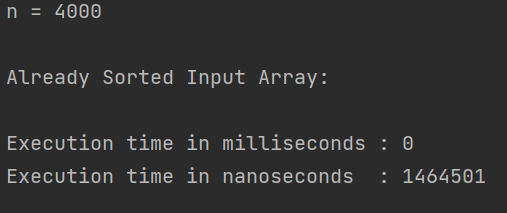
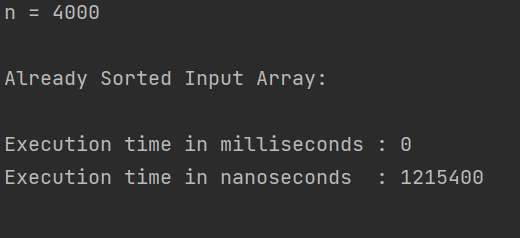
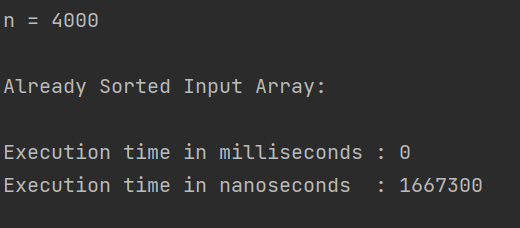
Heap Sort Special Case 1 for Small Input (Sorted Array):

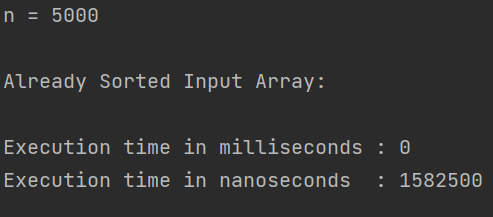
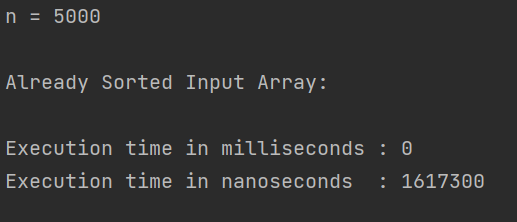
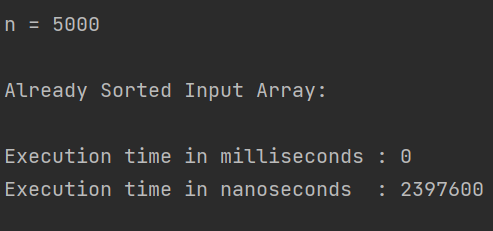
Table: Input size, and time in nanoseconds

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Input size | Run #1 | Run#2 | Run#3 | Average |
| 1K | 286800 | 274700 | 278400 | 279966 |
| 2K | 581500 | 762301 | 591801 | 645200 |
| 4K | 1667300 | 1215400 | 1464501 | 1449067 |
| 5K | 2397600 | 1617300 | 1582500 | 1865800 |



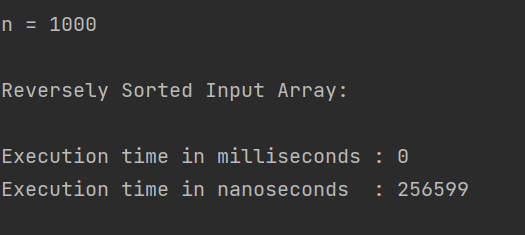
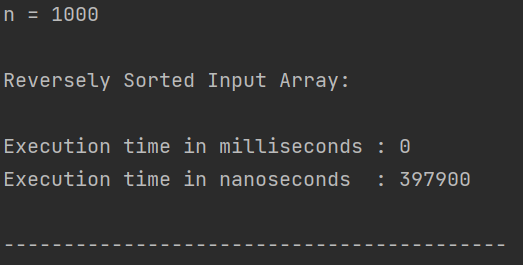
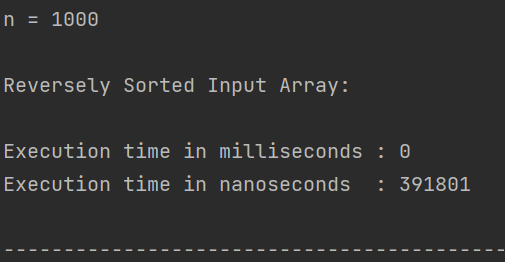


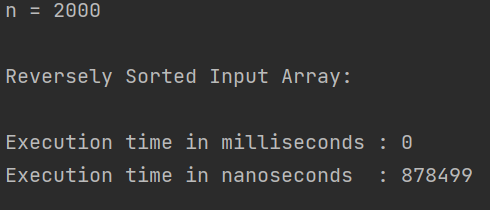
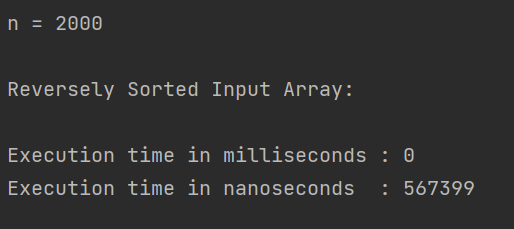
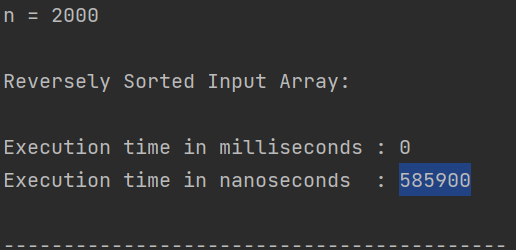


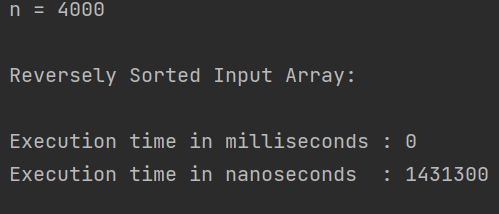
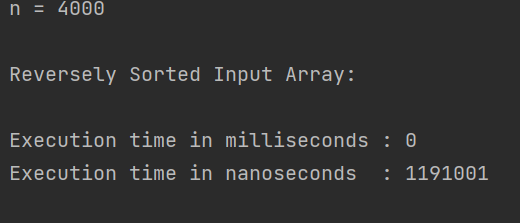
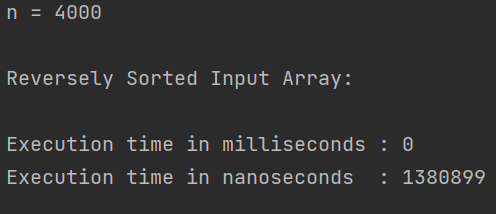


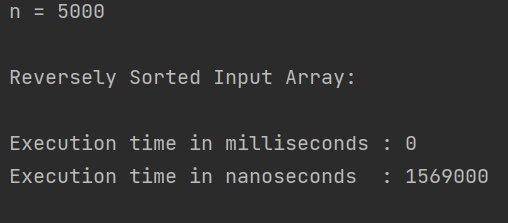
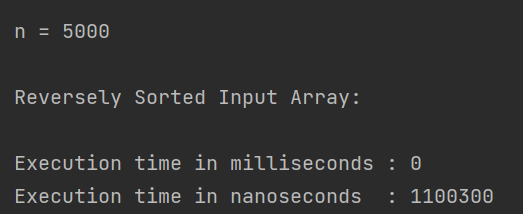
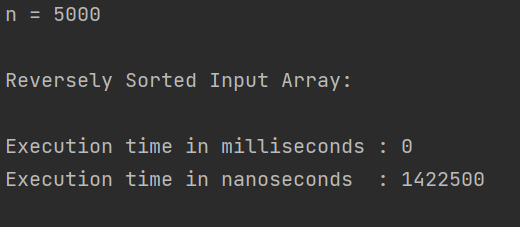
Heapsort Special Case #2 (Reverse sorted array)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Input Size | Run #1 | Run#2 | Run#3 | Average |
| 1K | 391801 | 397900 | 256599 | 348766 |
| 2K | 585900 | 567399 | 878499 | 677266 |
| 4K | 1380899 | 1191001 | 1431300 | 1334400 |
| 5K | 1422500 | 1100300 | 1569000 | 1363933 |









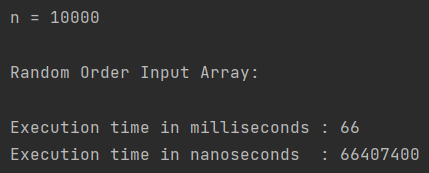
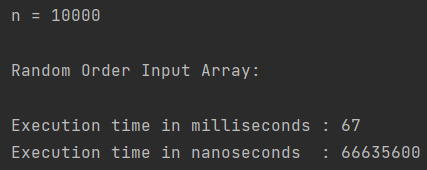
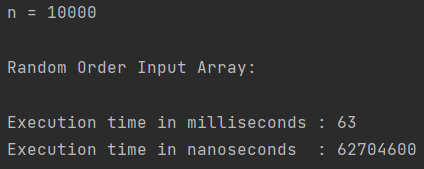
**Medium Sample Size Execution Results Graphs ( > = 10 K || <= 50 K):**

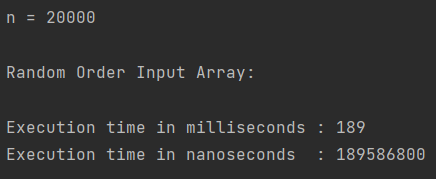
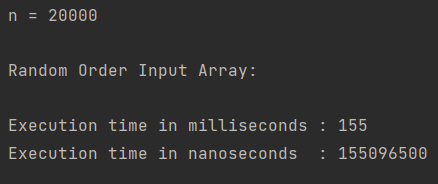
Medium size Random Order Input Graph Analysis:

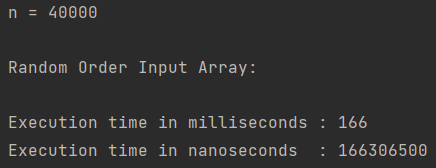
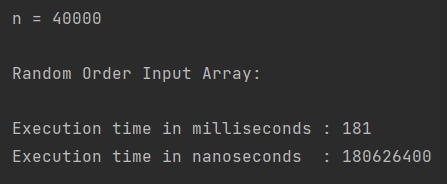
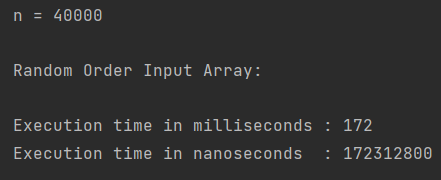
For medium size random order input we have a similar trend with small size random order inputs. Insertion sort has a worst and average running time of O(n^2).

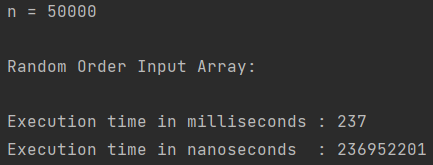
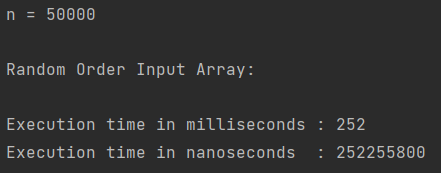
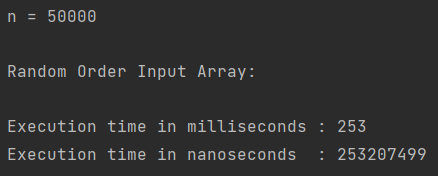
Insertion Sort (Random Order Input):

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Input Size n | Run # 1 | Run # 2 | Run #3 | Average |
| 10 K | 66407400 | 66635600 | 62704600 | 65249200 |
| 20 K | 189586800 | 155096500 | 150257701 | 164980333.7 |
| 40 K | 166306500 | 180626400 | 172312800 | 173081900 |
| 50 K | 236952201 | 252255800 | 253207499 | 247471833.3 |

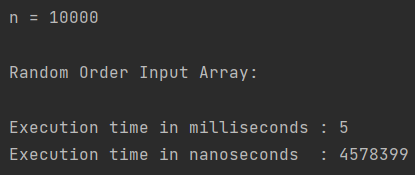
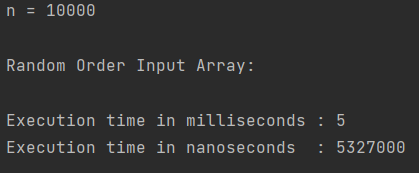
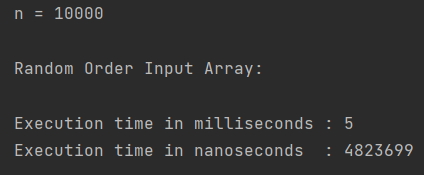
  

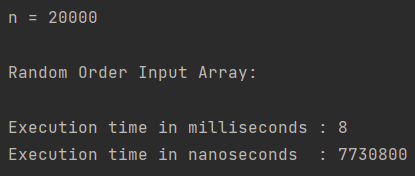
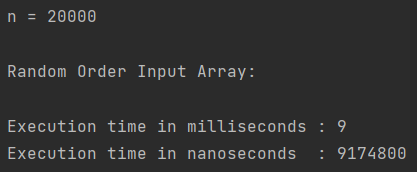
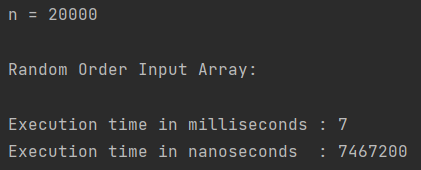
  

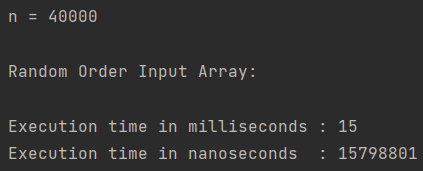
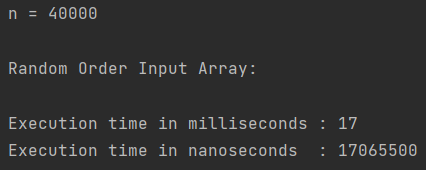
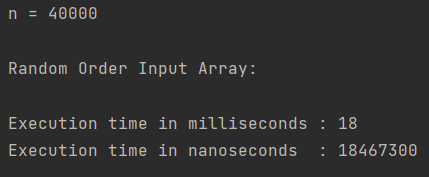
  

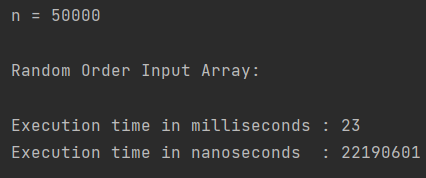
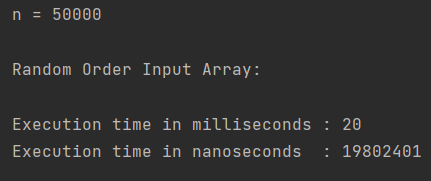
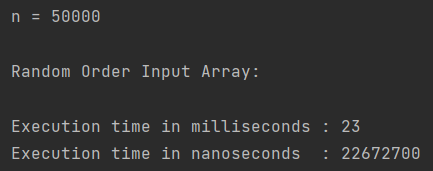
Merge Sort (Random Order Input):

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Input Size n | Run # 1 | Run # 2 | Run #3 | Average |
| 10 K | 4578399 | 5327000 | 4823699 | 4909699.333 |
| 20 K | 7730800 | 9174800 | 7467200 | 8124266.667 |
| 40 K | 15798801 | 17065500 | 18467300 | 17110533.67 |
| 50 K | 22190601 | 19802401 | 22672700 | 21555234 |

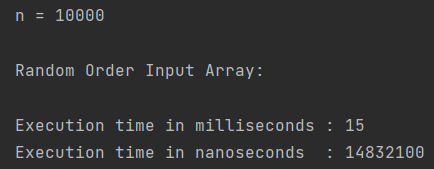
  

In Place Quick Sort (Random Order Input):

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Input Size n | Run # 1 | Run # 2 | Run #3 | Average |
| 10 K | 14832100 | 16436200 | 17748101 | 16338800.33 |
| 20 K | 31563300 | 24549100 | 35339600 | 30484000 |
| 40 K | 37954800 | 34556901 | 32645000 | 35052233.67 |
| 50 K | 51342801 | 43340801 | 43103100 | 45928900.67 |

 Text

Description automatically generated Text

Description automatically generated

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

Heap Sort:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Input Size | Run #1 | Run #2 | Run #3 | Average |
| 10K | 4029200 | 4091600 | 4777401 | 4299400 |
| 20K | 8848400 | 9660800 | 8845000 | 9118066 |
| 40K | 16673299 | 13945401 | 13375900 | 14664866 |
| 50L | 18475900 | 16046799 | 17270100 | 17264266 |

Text

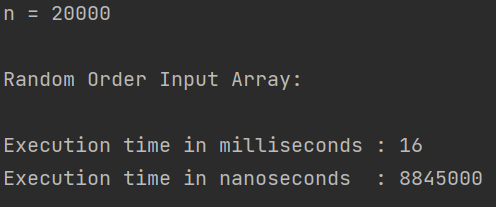
Description automatically generatedText

Description automatically generatedText

Description automatically generated

Text

Description automatically generatedText

Description automatically generated

Text

Description automatically generatedText

Description automatically generatedText

Description automatically generated

Text

Description automatically generatedU,{62454c28-6f2d-40f5-b2b9-bc73e65ac45a}{128},3.125,3.125Text

Description automatically generated

Medium size Already Sorted Input Graph Analysis:

For medium size already sorted input the in-place quicksort is our worst performer. It runs at its worst-case runtime O(n^2) when the input array is already sorted. This graph is similar to the smaller input size. As the input grows however, heapsort seams to do worse. Insertion sort runs best in this case.

Insertion Sort (Already Sorted Input):

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Input Size n | Run # 1 | Run # 2 | Run #3 | Average |
| 10 K | 101299 | 72400 | 22500 | 65399.66667 |
| 20 K | 149100 | 138200 | 159500 | 148933.3333 |
| 40 K | 779800 | 694401 | 570400 | 681533.6667 |
| 50 K | 989200 | 719200 | 575600 | 761333.3333 |

**Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated**

 Text

Description automatically generated Text

Description automatically generated

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

Merge Sort (Already Sorted Input):

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Input Size n | Run # 1 | Run # 2 | Run #3 | Average |
| 10 K | 2027299 | 2899999 | 1751300 | 2226199.333 |
| 20 K | 3494399 | 4707500 | 3600600 | 3934166.333 |
| 40 K | 5464599 | 6144101 | 6090900 | 5899866.667 |
| 50 K | 6907700 | 6239800 | 7672600 | 6940033.333 |

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

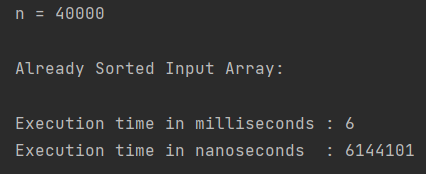
Text

Description automatically generated Text

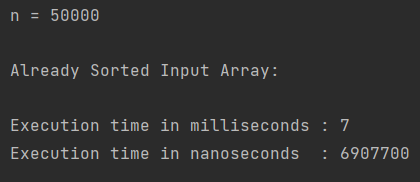
Description automatically generated Text

Description automatically generated

Text

Description automatically generated  Text

Description automatically generated

 Text

Description automatically generated Text

Description automatically generated

In Place Quick Sort (Already Sorted Input):

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Input Size n | Run # 1 | Run # 2 | Run #3 | Average |
| 10 K | 10729700 | 5354500 | 14431600 | 10171933.33 |
| 20 K | 7875700 | 71448700 | 6201400 | 28508600 |
| 40 K | 17416600 | 20701000 | 16501300 | 18206300 |
| 50 K | 23109800 | 23693400 | 20169900 | 22324366.67 |

**Text

Description automatically generated Text

Description automatically generated** **Text

Description automatically generated**

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

Heap Sort Special Case #1 (Already Sorted Array):

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Input Size | Run #1 | Run #2 | Run #3 | Average |
| 10K | 4091000 | 3138800 | 2924799 | 3384866 |
| 20K | 4427001 | 5389100 | 4599899 | 4805333 |
| 40K | 5969000 | 6925600 | 6642000 | 6512200 |
| 50K | 9294800 | 14312000 | 10909800 | 11505533 |

Text

Description automatically generatedText

Description automatically generatedText

Description automatically generated

Text

Description automatically generatedText

Description automatically generatedText

Description automatically generated

Text

Description automatically generatedText

Description automatically generatedText

Description automatically generated

Text

Description automatically generatedText

Description automatically generatedText

Description automatically generated

Medium size Reversely Sorted Input Graph Analysis:

In the medium size special case of reversely sorted input, the insertion sort continues to significantly do worse. In-place quicksort still struggles but when compared to insertion sort almost looks as good as merge and heap sort.

Insertion Sort (Reversely Sorted Input):

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Input Size n | Run # 1 | Run # 2 | Run #3 | Average |
| 10 K | 66161700 | 62165200 | 63279100 | 63868666.67 |
| 20 K | 251760100 | 267335800 | 253670600 | 257588833.3 |
| 40 K | 330769800 | 407943200 | 315208500 | 351307166.7 |
| 50 K | 474811200 | 482682800 | 498593800 | 485362600 |

**Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated**

Text

Description automatically generated Text

Description automatically generated Text

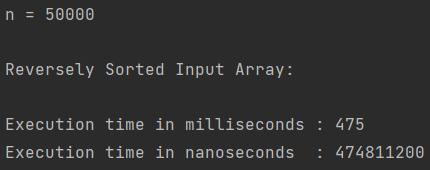
Description automatically generated

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

 Text

Description automatically generated Text

Description automatically generated

Merge Sort (Reversely Sorted Input):

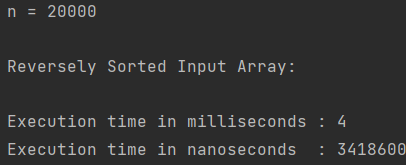
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Input Size n | Run # 1 | Run # 2 | Run #3 | Average |
| 10 K | 2605600 | 2317700 | 1577400 | 2166900 |
| 20 K | 3418600 | 3915100 | 3911300 | 3748333.333 |
| 40 K | 4331200 | 6136200 | 3976200 | 4814533.333 |
| 50 K | 5174700 | 5262900 | 4987200 | 5141600 |

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

 Text

Description automatically generated Text

Description automatically generated

Text

Description automatically generated Text

Description automatically generated 

 Text

Description automatically generated Text

Description automatically generated

In Place Quick Sort (Reversely Sorted Input):

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Input Size n | Run # 1 | Run # 2 | Run #3 | Average |
| 10 K | 2613700 | 2691300 | 5201500 | 3502166.667 |
| 20 K | 8031200 | 9000200 | 8840600 | 8624000 |
| 40 K | 14736200 | 14506600 | 16414500 | 15219100 |
| 50 K | 20821900 | 22757500 | 29407700 | 24329033.33 |

**Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated**

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

Text

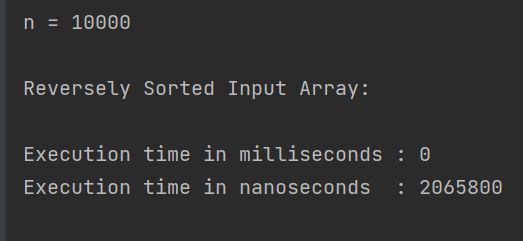
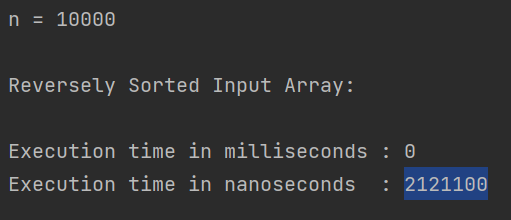
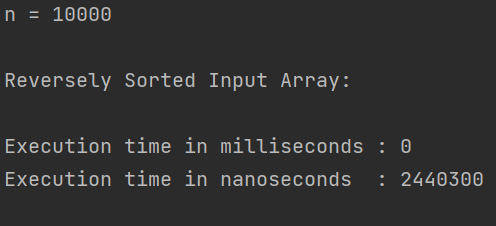
Description automatically generated Text

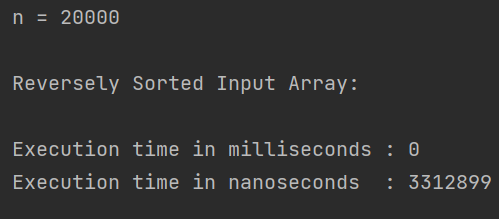
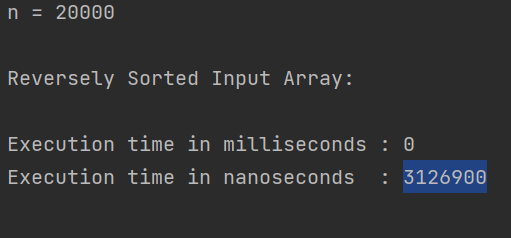
Description automatically generated Text

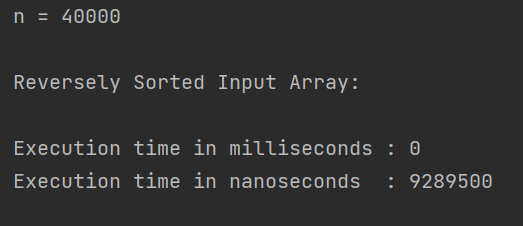
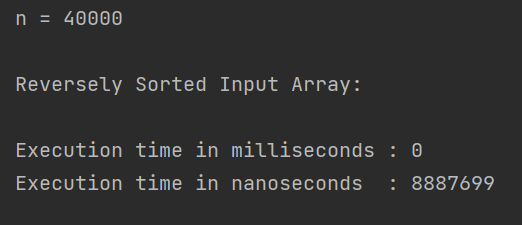
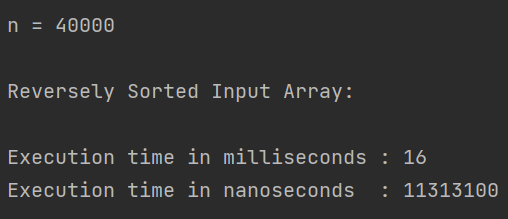
Description automatically generated

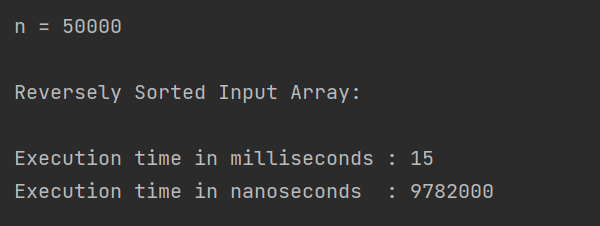
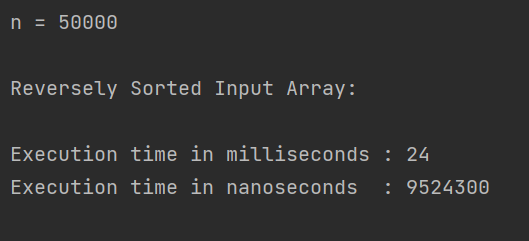
Heap Sort Special Case #2 (Reverse sorted array):

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Input Size | Run #1 | Run #2 | Run #2 | Average |
| 10K | 2440300 | 2121100 | 2065800 | 2209066 |
| 20K | 3126900 | 4074400 | 3312899 | 3504733 |
| 40K | 11313100 | 8887699 | 9289500 | 9830099 |
| 50K | 10903600 | 9524300 | 9782000 | 10069966 |







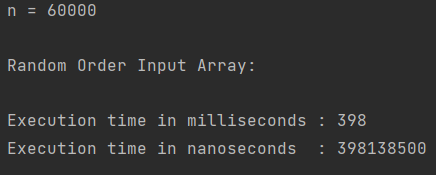


**Large Sample Size Execution Results Graphs ( > = 60 K || <= 1 M):**

Large size Random Order Input Graph Analysis:

Insertion Sort (Random Order Input):

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Input Size n | Run # 1 | Run # 2 | Run #3 | Average |
| 60 K | 398138500 | 347825500 | 366188700 | 370717566.7 |
| 80 K | 620923800 | 716013100 | 621233300 | 652723400 |
| 100 K | 940499300 | 1221745100 | 1247957400 | 1136733933 |
| 1 M |  |  |  |  |

 Text

Description automatically generated Text

Description automatically generated

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

Merge Sort (Random Order Input):

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Input Size n | Run # 1 | Run # 2 | Run #3 | Average |
| 60 K | 17630700 | 22908900 | 20737600 | 20425733.33 |
| 80 K | 38336000 | 31363900 | 26146100 | 31948666.67 |
| 100 K | 30154100 | 40828100 | 41385100 | 37455766.67 |
| 1 M | 191173900 | 242738000 | 196933900 | 210281933.3 |

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

In place Quick Sort (Random Order Input):

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Input Size n | Run # 1 | Run # 2 | Run #3 | Average |
| 60 K | 47540800 | 54512800 | 39919200 | 47324266.67 |
| 80 K | 58050400 | 70927200 | 59499900 | 62825833.33 |
| 100 K | 69283700 | 81541700 | 122896300 | 91240566.67 |
| 1 M | 1245538200 | 1771692400 | 1779052300 | 1598760967 |

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

Text

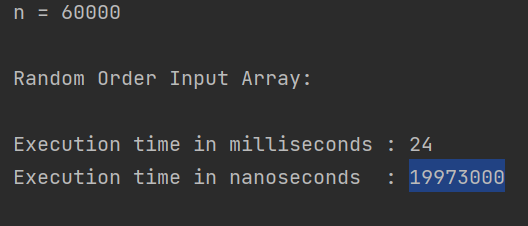
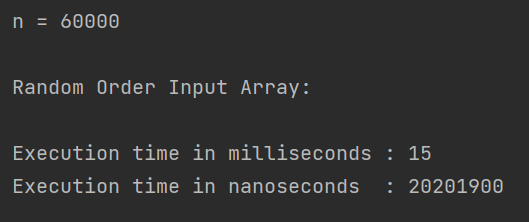
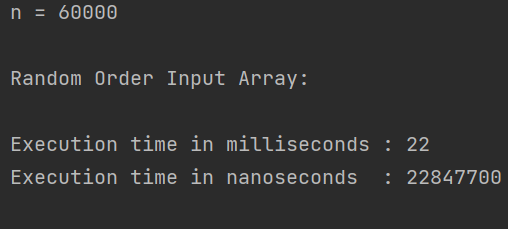
Description automatically generated Text

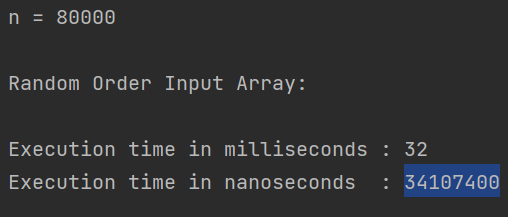
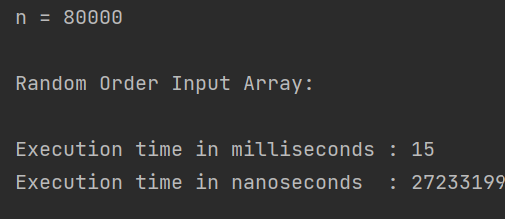
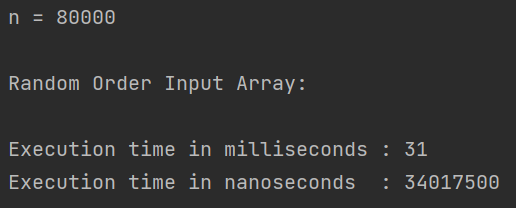
Description automatically generated Text

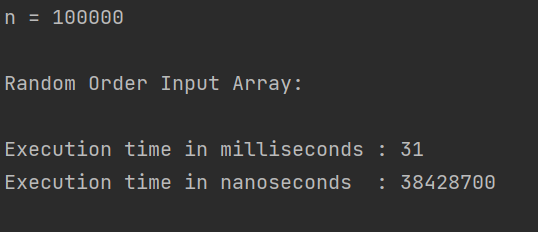
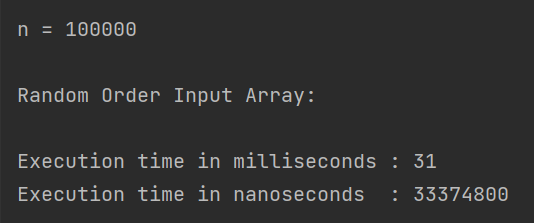
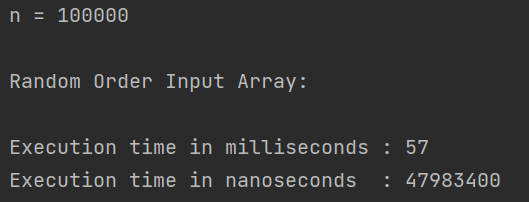
Description automatically generated

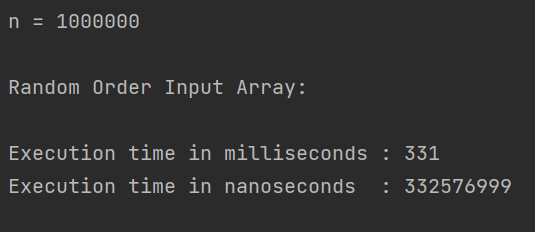
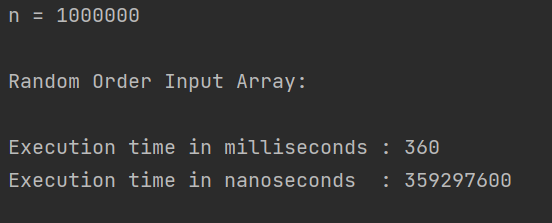
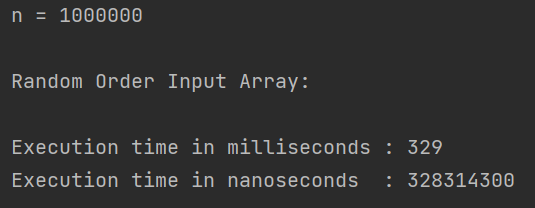
Heap Sort:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Input Size | Run #1 | Run #2 | Run #3 | Average |
| 60K | 22847700 | 20201900 | 19973000 | 21007533.33 |
| 80K | 34017500 | 27233199 | 34107400 | 31786033 |
| 100K | 47983400 | 33374800 | 38428700 | 39928966.67 |
| 1M | 328314300 | 359297600 | 332576999 | 340062966.3 |









Insertion Sort (Already Sorted Input):

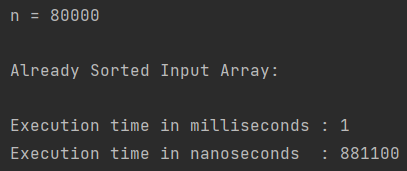
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Input Size n | Run # 1 | Run # 2 | Run #3 | Average |
| 60 K | 662200 | 755800 | 1203100 | 873700 |
| 80 K | 881100 | 226300 | 177900 | 428433.3333 |
| 100 K | 953700 | 188300 | 313700 | 485233.3333 |
| 1 M | 5486200 | 8725900 | 6046300 | 6752800 |

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

 Text

Description automatically generated Text

Description automatically generated

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

Merge Sort (Already Sorted Input):

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Input Size n | Run # 1 | Run # 2 | Run #3 | Average |
| 60 K | 15345800 | 13686200 | 14809500 | 14613833.33 |
| 80 K | 19029500 | 18122600 | 21318400 | 19490166.67 |
| 100 K | 21528200 | 25879900 | 27763400 | 25057166.67 |
| 1 M | 83941800 | 99615300 | 106467100 | 96674733.33 |

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

In place Quick Sort (Already Sorted Input):

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Input Size n | Run # 1 | Run # 2 | Run #3 | Average |
| 60 K | 70601900 | 73674900 | 51757500 | 65344766.67 |
| 80 K | 102043300 | 95630400 | 80883700 | 92852466.67 |
| 100 K | 105932100 | 102476200 | 93478900 | 100629066.7 |
| 1 M | 1374436900 | 2374246500 | 1449506000 | 1732729800 |

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

Text

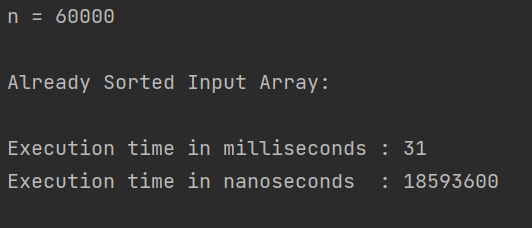
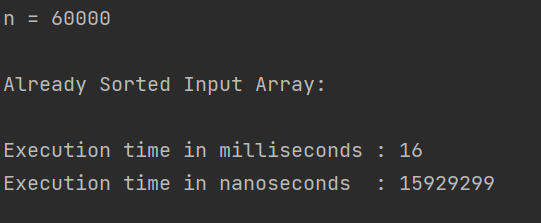
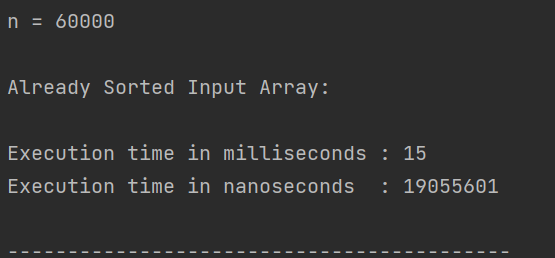
Description automatically generated Text

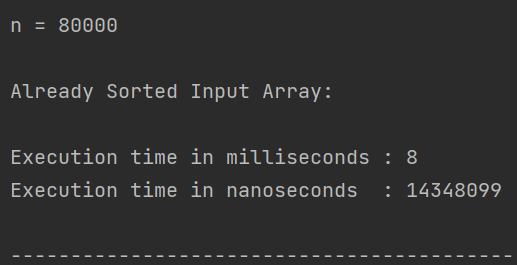
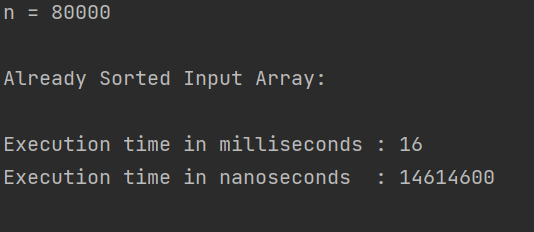
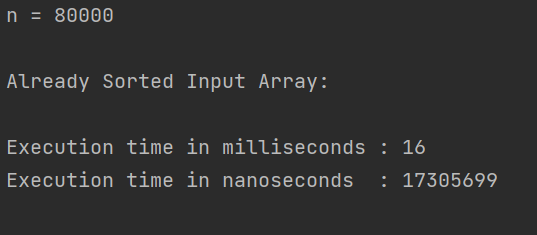
Description automatically generated Text

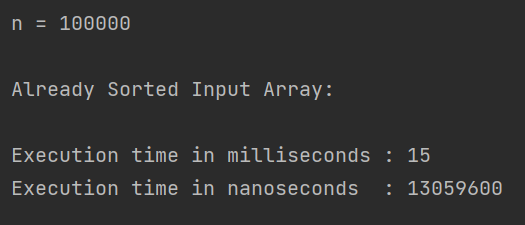
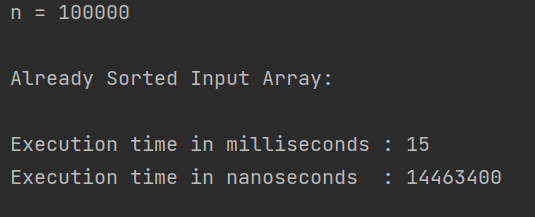
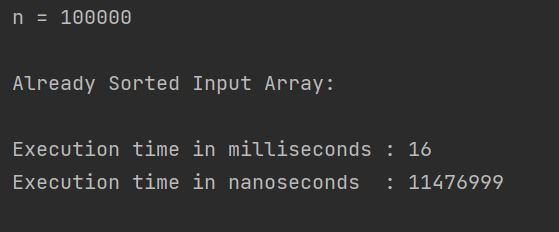
Description automatically generated

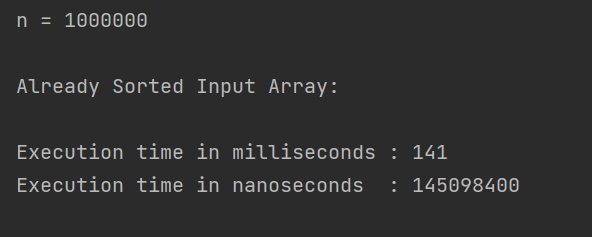
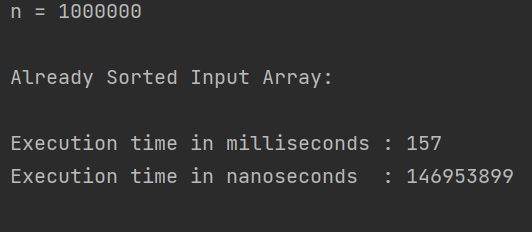
Heap Sort Special Case #1: Already Sorted array

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Input Size | Run #1 | Run #2 | Run #2 | Average |
| 60K | 19055601 | 15929299 | 18593600 | 17859500 |
| 80K | 17305699 | 14614600 | 14348099 | 15422799 |
| 100K | 11476999 | 14463400 | 13059600 | 12999999 |
| 1M | 146953899 | 145098400 | 150769900 | 147607399 |









Insertion Sort (Reversely Sorted Input):

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Input Size n | Run # 1 | Run # 2 | Run #3 | Average |
| 60 K | 668815200 | 733397000 | 735850000 | 712687400 |
| 80 K | 1308373400 | 1492046300 | 1239832800 | 1346750833 |
| 100 K | 2121994600 | 1933651000 | 2763472900 | 2273039500 |
| 1 M | 203586661900 | 207244867600 | 204927491800 | 205253007100 |

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

Merge Sort (Reversely Sorted Input):

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Input Size n | Run # 1 | Run # 2 | Run #3 | Average |
| 60 K | 17469000 | 14771500 | 13604600 | 15281700 |
| 80 K | 19188400 | 24989700 | 20422700 | 21533600 |
| 100 K | 24658800 | 37225900 | 25028200 | 28970966.67 |
| 1 M | 132468100 | 105238400 | 113619400 | 117108633.3 |

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

Text

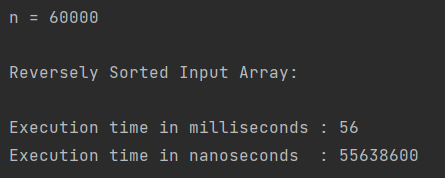
Description automatically generated Text

Description automatically generated Text

Description automatically generated

In place Quick Sort (Reversely Sorted Input):

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Input Size n | Run # 1 | Run # 2 | Run #3 | Average |
| 60 K | 55638600 | 51065900 | 58923800 | 55209433.33 |
| 80 K | 107204400 | 90897500 | 120764400 | 106288766.7 |
| 100 K | 103132000 | 112217400 | 119809000 | 111719466.7 |
| 1 M | 1328513000 | 1327086900 | 1923170300 | 1526256733 |

 Text

Description automatically generated Text

Description automatically generated

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

Text

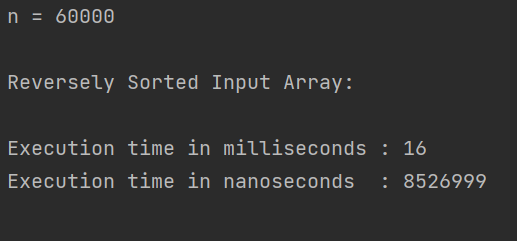
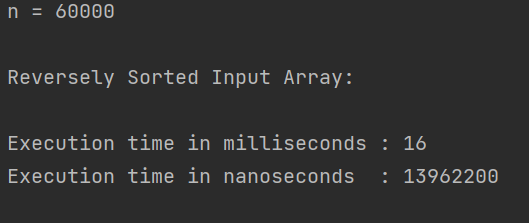
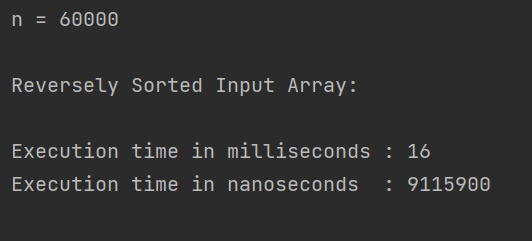
Description automatically generated Text

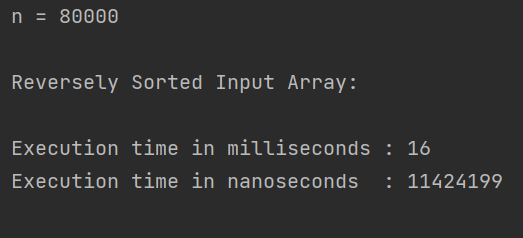
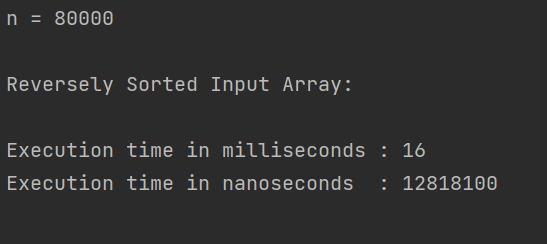
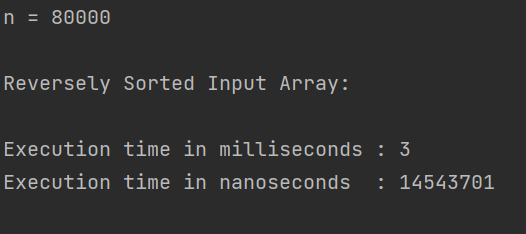
Description automatically generated Text

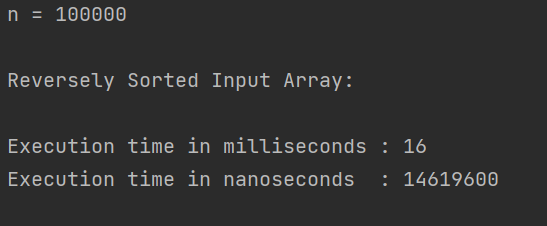
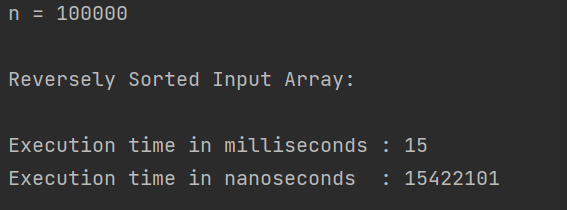
Description automatically generated

Heap Sort (Reverse Sorted array):

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Input Size | Run #1 | Run #2 | Run #3 | Average |
| 60K | 9115900 | 13962200 | 8526999 | 10535033 |
| 80K | 14543701 | 12818100 | 11424199 | 12928666 |
| 100K | 15422101 | 14948900 | 14619600 | 14996867 |







**Extra Large Sample Size Execution Results Graphs ( > = 2 M || <= 50 M):**

// in place quicksort

Text

Description automatically generated