Radix Sort

Борейк Али Акиль Али

1 About Radix sort

What is the Radix sort ?

The lower bound for Comparison based sorting algorithm (Merge Sort, Heap Sort, Quick-Sort .. etc.) is Ω(n Log n), i.e., they cannot do better than n Log n.
Counting sort is a linear time sorting algorithm that sort in O(n +k) time when elements are in the range from 1 to k.
What if the elements are in the range from 1 to n2?
We can't use counting sort because counting sort will take O(n2) which is worse than comparison-based sorting algorithms. Can we sort such an array in linear time?
Radix Sort is the answer. The idea of Radix Sort is to do digit by digit sort starting from least significant digit to most significant digit. Radix sort uses counting sort as a subroutine to sort.
The Radix Sort Algorithm
1. Do following for each digit i where i varies from least significant digit to the most significant digit.
Sort input array using counting sort (or any stable sort) according to the it's digit.

What is the running time of Radix Sort ?

Let there be d digits in input integers. Radix Sort takes O(d*(n+b)) time where b is the base for representing numbers, for example, for the decimal system, b is 10. What is the value of d? If k is the maximum possible value, then d would be O(logb(k)). So overall time complexity is O((n+b) * logb(k)). Which looks more than the time complexity of comparison-based sorting algorithms for a large k. Let us first limit k. Let k <= nc where c is a constant. In that case, the complexity becomes O(nLogb(n)). But it still doesn't beat comparison-based sorting algorithms.

What if we make the value of b larger?. What should be the value of b to make the time complexity linear? If we set b as n, we get the time complexity as O(n). In other words, we can sort an array of integers with a range from 1 to nc if the numbers are represented in base n (or every digit takes log2(n) bits).

Is Radix Sort preferable to Comparison based sorting algorithms like Quick-Sort?

If we have log2n bits for every digit, the running time of Radix appears to be better than effectively. Also, Radix sort uses counting sort as a subroutine and counting sort takes extra space to sort numbers.

Recommended: Please try your approach on {IDE} first, before moving on to the solution.

2

Implementation of Radix Sort

In java

https://bitbucket.org/AliBoraik/boraik_11-013/src/master/homework/src/Radix%20sort/Radix_only_algorithms.java

3 Example of a logarithm

Consider this input array

170	45	75	90	802	24	2	66

First consider the one's place

Example of a logarithm

Consider this input array

170	45	75	90	802	24	2	66

170	90	802	2	24	45	75	66

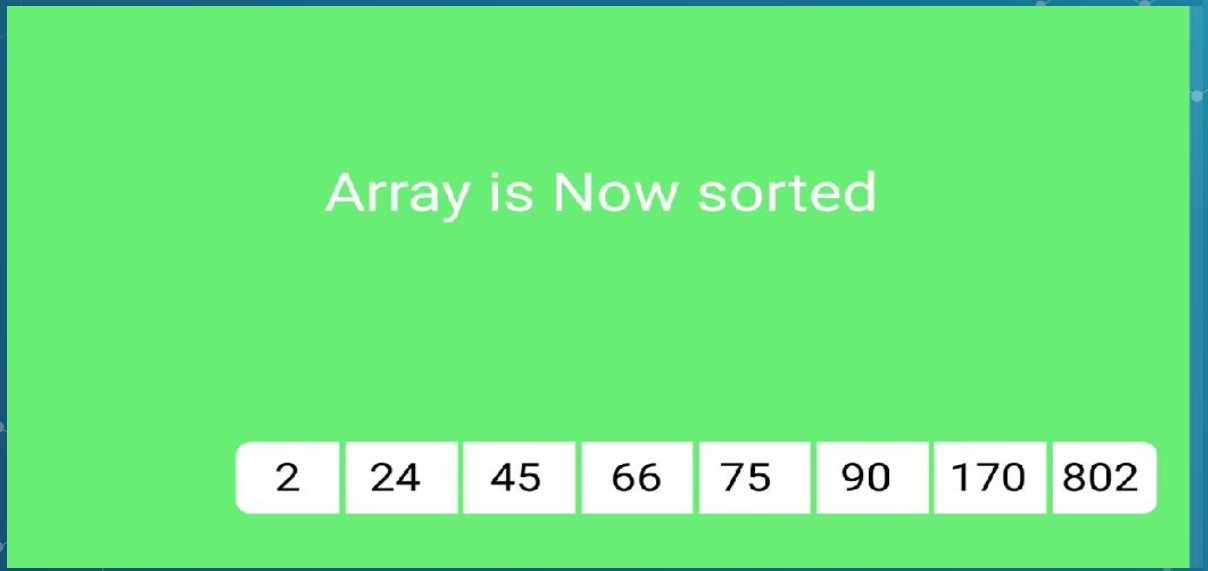
Example of a logarithm

Consider this input array

170	45	75	90	802	24	2	66
170	90	802	2	24	45	75	66
802	2	24	45	66	170	75	90

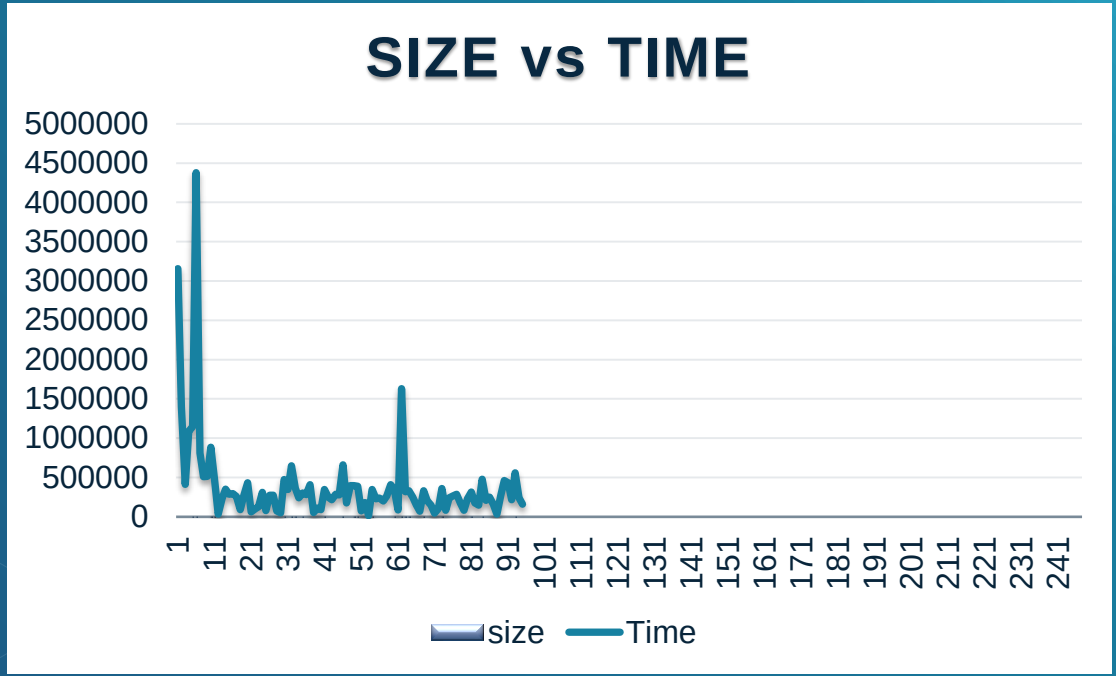
Now consider the 100's place.

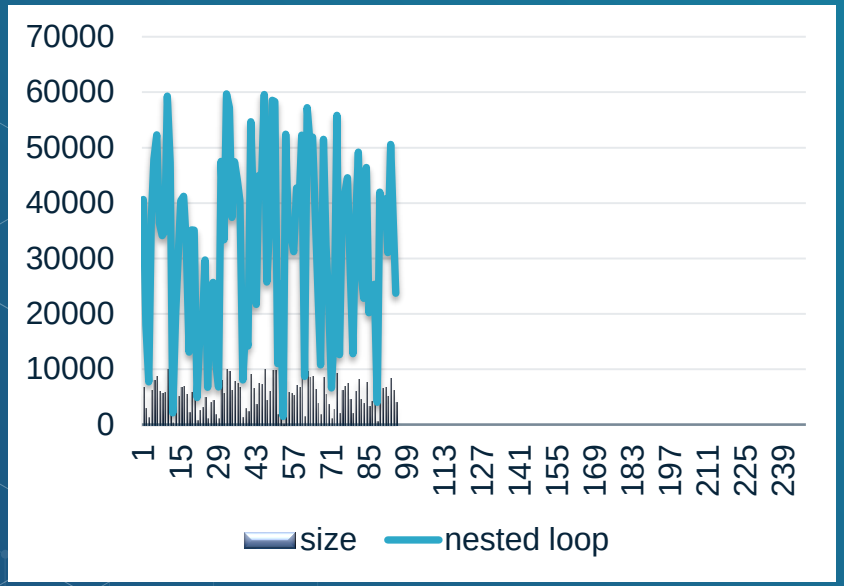
Example of a logarithm

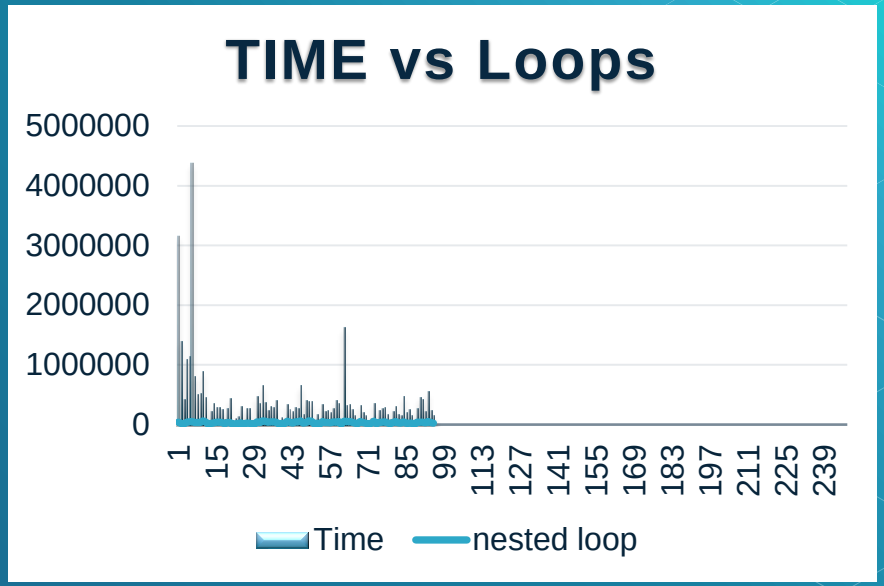


4

Comparison of data (attached in the excel)


SIZE vs TIME
5000000
4500000
4000000
3500000
3000000
2500000
2000000
1500000
1000000
500000
0
1
11
21
31
41
51
61
71
81
91
101
111
121
131
141
151
161
171
181
191
201
211
221
231
241
size
Time


70000
60000
50000
40000
30000
20000
10000
0
1
15
29
43
57
71
85
99
113
127
141
155
169
183
197
211
225
239
size
nested loop


TIME vs Loops
5000000
4000000
3000000
2000000
1000000
0
1
15
29
43
57
71
85
99
113
127
141
155
169
183
197
211
225
239
Time
nested loop