Does Heapsort work in time o(n log n) in the best case?

Is it possible for Heapsort to work in time $o(n \log n)$ on certain inputs?

For example in case of Insertion sort it is possible, however when it comes to Quickssort it is not possible. What about Heapsort? I tried to find an input array such that Heapsort will be working in $o(n \log n)$.

I ask you is it possible? Some permutation? The same elements?

algorithms runtime-analysis sorting lower-bounds heap-sort

edited Jan 20 '16 at 11:06
Raphael ♦
55.4k ● 21 ● 135 ● 296

asked Jan 19 '16 at 21:40
user40545
303 ● 1 ● 11

1 Answer

The paper: The Analysis of Heapsort by Schaffer and Sedgewick shows that

> Theorem 1: Heapsort requires that at least $\frac{1}{2} n \lg n - O(n)$ data moves for any heap composed of distinct keys.

and that

> Theorem 3: The *average* number of data moves required to Heapsort a random permutation of n distinct keys is $\sim n \lg n$.

It also mentions that

> If equal keys are allowed, the best case is clearly linear [3]: consider the case of a heap with all keys equal.

The citation to [3] here is TechReport: On Heapsort and its Dependence on Input Data.

edited Jan 20 '16 at 13:38

answered Jan 20 '16 at 2:19
hengxin
6,662 ● 13 ● 45

Add Another Answer