Lower bound for comparison based sorting algorithms

The problem of sorting can be viewed as following.

Input: A sequence of *n* numbers <a_1, a_2, . . . , a_n>.
Output: A permutation (reordering) <a'_1, a'_2, . . . , a'_n> of the input sequence such that a'_1 <= a'_2 ….. <= a'_n.

A sorting algorithm is comparison based if it uses comparison operators to find the order between two numbers. Comparison sorts can be viewed abstractly in terms of decision trees. A decision tree is a full binary tree that represents the comparisons between elements that are performed by a particular sorting algorithm operating on an input of a given size. The execution of the sorting algorithm corresponds to tracing a path from the root of the decision tree to a leaf. At each internal node, a comparison a_i □ a_j is made. The left subtree then dictates subsequent comparisons for a_i □ a_j, and the right subtree dictates subsequent comparisons for $a_i > a_j$. When we come to a leaf, the sorting algorithm has established the ordering. So we can say following about the decison tree.

1) Each of the *n*! permutations on *n* elements must appear as one of the leaves of the decision tree for the sorting algorithm to sort properly.

2) Let x be the maximum number of comparisons in a sorting algorithm. The maximum height of the decison tree would be x. A tree with maximum height x has at most 2^x leaves.

After combining the above two facts, we get following relation.

```
n!  <= 2^x
```


Interview Experiences
Advanced Data Structures
Dynamic Programming
Greedy Algorithms
Backtracking
Pattern Searching
Divide & Conquer
Mathematical Algorithms
Recursion
Geometric Algorithms

Taking Log on both sides.
$\log_2 n!$ <= x

Since $\log_2 n!$ = $\Theta(nLogn)$, we can say
x = $\Omega(nLog_2n)$

Therefore, any comparison based sorting algorithm must make at least $\Omega(nLog_2n)$ comparisons to sort the input array, and Heapsort and merge sort are asymptotically optimal comparison sorts.

References:

Introduction to Algorithms, by Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest and Clifford Stein

Related Tpoics:

- Backtracking | Set 8 (Solving Cryptarithmetic Puzzles)
- Tail Recursion
- Find if two rectangles overlap

Popular Posts

- All permutations of a given string
- Memory Layout of C Programs
- Understanding “extern” keyword in C
- Median of two sorted arrays
- Tree traversal without recursion and without stack!
- Structure Member Alignment, Padding and Data Packing
- Intersection point of two Linked Lists
- Lowest Common Ancestor in a BST.
- Check if a binary tree is BST or not
- Sorted Linked List to Balanced BST

- Find if two rectangles overlap
- Analysis of Algorithm | Set 4 (Solving Recurrences)
- Print all possible paths from top left to bottom right of a mXn matrix
- Generate all unique partitions of an integer
- Russian Peasant Multiplication
- Closest Pair of Points | O(nlogn) Implementation

2 Tweet 0 0

Writing code in comment? Please use **ideone.com** and share the link here.

7 Comments **GeeksforGeeks**

Sort by Newest

Join the discussion…

Kanhaiya Kumawat • 4 months ago

there is typo in the last line: "Therefore, any comparison based sorting algorith
sort the input array, and Heapsort and merge sort are asymptotically optimal c
its nLogn rather than Logn.

GeeksforGeeks Mod Kanhaiya Kumawat • 3 months ago

Thanks for pointing this out. We have corrected the typo.

wgpshashank • 3 years ago

More Info ..

http://www.it-c.dk/courses/ITM/F2003/Sorting.pdf

lovocas · 3 years ago

" A decision tree is a full binary tree"

I am confused by this statement. :-<

A full binary tree is a tree has 2^(k+1)-1 nodes ?k is the heright,the root&#039s

I think desision tree is just a binary tree, whose nodes has either 2 or zero nod

GeeksforGeeks → lovocas · 3 years ago

@lovocas: The statement looks correct. Please see the following Wiki

A full binary tree (sometimes proper binary tree or 2-tree or strictly bina other than the leaves has two children.

To avoid confusion, we have updated the post and added the Wiki link

Yang → GeeksforGeeks · 7 months ago

What if I write a "really bad" algorithm and it asks the same que answer every time. Then, the Decision tree will still be full binar

lovocas → GeeksforGeeks · 3 years ago

oh, thanks very much , got it!

Recent Comments

Aman Hi, Why arent we checking for conditions...

Write a C program to Delete a Tree. · 32 minutes ago

kzs please provide solution for the problem...

Backtracking | Set 2 (Rat in a Maze) · 36 minutes ago

Sanjay Agarwal bool tree::Root_to_leaf_path_given_sum(tree...

Root to leaf path sum equal to a given number · 1 hour ago

GOPI GOPINATH @admin Highlight this sentence "We can easily...

Count trailing zeroes in factorial of a number · 1 hour ago

newCoder3006 If the array contains negative numbers also. We...

Find subarray with given sum · 1 hour ago

newCoder3006 Code without using while loop. We can do it...

Find subarray with given sum · 1 hour ago