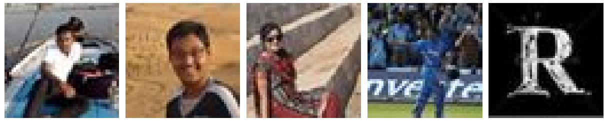

Sorting

Question 1

What is recurrence for worst case of QuickSort and what is the time complexity in Worst case?

A Recurrence is T(n) = T(n-2) + O(n) and time complexity is O(n^2)

B Recurrence is T(n) = T(n-1) + O(n) and time complexity is O(n^2)

C Recurrence is T(n) = 2T(n/2) + O(n) and time complexity is O(nLogn)

D Recurrence is T(n) = T(n/10) + T(9n/10) + O(n) and time complexity is O(nLogn)

Discuss it

Question 2

Suppose we have a O(n) time algorithm that finds median of an unsorted array. Now consider a QuickSort implementation where we first find median using the above algorithm, then use median as pivot. What will be the worst case time complexity of this modified QuickSort.

A O(n^2 Logn)

B O(n^2)

C O(n Logn Logn)

D O(nLogn)

Discuss it

Question 3

Which of the following is not a stable sorting algorithm in its typical implementation.

A Insertion Sort

B Merge Sort

C Quick Sort

D Bubble Sort

Discuss it

Question 4

Which of the following sorting algorithms in its typical implementation gives best

Categories

Articles (28)
- C (4)
- C++ (2)
- Data Structures (13)
- DBMS (1)
- Operating Systems (1)
- Searching and Sorting (7)

Programs (7)

Quizzes (1,392)
- Aptitude (1)
- Computer Science Quizzes (1,391)
  - Algorithms (146)
  - C (203)
  - C++ (123)
  - Data Structures (131)
  - GATE (709)
  - Java (51)
  - Operating Systems (28)

performance when applied on an array which is sorted or almost sorted (maximum 1 or two elements are misplaced).

A Quick Sort

B Heap Sort

C Merge Sort

D Insertion Sort

Discuss it

Question 5

Given an unsorted array. The array has this property that every element in array is at most k distance from its position in sorted array where k is a positive integer smaller than size of array. Which sorting algorithm can be easily modified for sorting this array and what is the obtainable time complexity?

A Insertion Sort with time complexity O(kn)

B Heap Sort with time complexity O(nLogk)

C Quick Sort with time complexity O(kLogk)

D Merge Sort with time complexity O(kLogk)

Discuss it

Recent Discussions

Sumit Khatri this is the sorting technique which can work...

Insertion Sort · 7 hours ago

Sumit Khatri no, quick sort requires more swaps than...

Selection Sort · 7 hours ago

Sumit Khatri yes, it is the only sorting technique which...

Selection Sort · 7 hours ago

Question 6

Consider a situation where swap operation is very costly. Which of the following sorting algorithms should be preferred so that the number of swap operations are minimized in general?

A Heap Sort

B Selection Sort

C Insertion Sort

D Merge Sort

Discuss it

Question 7

Which of the following is not true about comparison based sorting algorithms?

A The minimum possible time complexity of a comparison based sorting algorithm is O(nLogn) for a random input array

B Any comparison based sorting algorithm can be made stable by using position as a criteria when two elements are compared

C Counting Sort is not a comparison based sorting algortihm

D Heap Sort is not a comparison based sorting algorithm.

Discuss it

Sudhakar Mishra I think it should be 2n + 1

Data Structures | Binary Trees | Question 12 · 7 hours ago

Sudhakar Mishra (2n)!/((n+1)!*n!)

Data Structures | Binary Trees | Question 6 · 1 day ago

Sudhakar Mishra Always Y will be more than one because after...

Data Structures | Stack | Question 7 · 1 day ago

Question 8

Suppose we are sorting an array of eight integers using quicksort, and we have just finished the first partitioning with the array looking like this: 2 5 1 7 9 12 11 10 Which statement is correct?

A The pivot could be either the 7 or the 9.

B The pivot could be the 7, but it is not the 9

C The pivot is not the 7, but it could be the 9

D Neither the 7 nor the 9 is the pivot.

Discuss it

Question 9

Suppose we are sorting an array of eight integers using heapsort, and we have just finished some heapify (either maxheapify or minheapify) operations. The array now looks like this: 16 14 15 10 12 27 28 How many heapify operations have been performed on root of heap?

A 1

B 2

C 3 or 4

D 5 or 6

Discuss it

Question 10

What is the best time complexity of bubble sort?

A N^2

B NlogN

C N

D N(logN)^2

Discuss it

Question 11

You have to sort 1 GB of data with only 100 MB of available main memory. Which sorting technique will be most appropriate?

A Heap sort

B Merge sort

C Quick sort

D Insertion sort

Discuss it

Question 12

What is the worst case time complexity of insertion sort where position of the data to be inserted is calculated using binary search?

A N

B NlogN

C N^2

D N(logN)^2

Discuss it

Question 13

The tightest lower bound on the number of comparisons, in the worst case, for comparison-based sorting is of the order of

A N

B N^2

C NlogN

D N(logN)^2

Discuss it

Question 14

In a modified merge sort, the input array is splitted at a position one-third of the length(N) of the array. What is the worst case time complexity of this merge sort?

A N(logN base 3)

B N(logN base 2/3)

C N(logN base 1/3)

D N(logN base 3/2)

Discuss it

Question 15

Which sorting algorithm will take least time when all elements of input array are identical? Consider typical implementations of sorting algorithms.

A Insertion Sort

B Heap Sort

C Merge Sort

D Selection Sort

Discuss it

Question 16

A list of n string, each of length n, is sorted into lexicographic order using the merge-sort algorithm. The worst case running time of this computation is (A) O (n log n (B) O (n^2 log n) (C) O (n^2 + log n) (D) □

A A

B B

C C

D D

Discuss it

Question 17

In quick sort, for sorting n elements, the (n/4)th smallest element is selected as pivot using an O(n) time algorithm. What is the worst case time complexity of the quick sort? (A) □(n) (B) □(nLogn) (C) □(n^2) (D) □(n^2 log n)

A A

B B

C C

D D

Discuss it

Question 18

Consider the Quicksort algorithm. Suppose there is a procedure for finding a pivot element which splits the list into two sub-lists each of which contains at least one-fifth of the elements. Let T(n) be the number of comparisons required to sort n elements. Then

A $T(n) <= 2T(n/5) + n$

B $T(n) <= T(n/5) + T(4n/5) + n$

C $T(n) <= 2T(4n/5) + n$

D $T(n) <= 2T(n/2) + n$

Discuss it

Question 19

Which of the following sorting algorithms has the lowest worst-case complexity?

A Merge Sort

B Bubble Sort

C Quick Sort

D Selection Sort

Discuss it

Question 20

Which sorting algorithms is most efficient to sort string consisting of ASCII characters?

A Quick sort

B Heap sort

C Merge sort

D Counting sort

Discuss it

Question 21

The number of elements that can be sorted in time using heap sort is

```
(A)
(B)
(C)
(d)
```

A A

B B

C C

D D

Discuss it

Question 22

Which of the following is true about merge sort?

A Merge Sort works better than quick sort if data is accessed from slow sequential memory.

B Merge Sort is stable sort by nature

C Merge sort outperforms heap sort in most of the practical situations.

D All of the above.

Discuss it

Question 23

Given an array where numbers are in range from 1 to n^6, which sorting algorithm can be used to sort these number in linear time?

A Not possible to sort in linear time

B Radix Sort

C Counting Sort

D Quick Sort

Discuss it

Question 24

In quick sort, for sorting n elements, the (n/4)th smallest element is selected as pivot using an O(n) time algorithm. What is the worst case time complexity of the quick sort? <pre> (A) θ(n) (B) θ(nLogn) (C) θ(n^2) (D) θ(n^2 log n) </pre>

A A

B B

C C

D D

Discuss it

Question 25

Consider the Quicksort algorithm. Suppose there is a procedure for finding a pivot element which splits the list into two sub-lists each of which contains at least one-fifth of the elements. Let T(n) be the number of comparisons required to sort n elements. Then

A T(n) <= 2T(n/5) + n

B $T(n) <= T(n/5) + T(4n/5) + n$

C $T(n) <= 2T(4n/5) + n$

D $T(n) <= 2T(n/2) + n$

Discuss it

Question 26

Let P be a QuickSort Program to sort numbers in ascending order using the first element as pivot. Let t1 and t2 be the number of comparisons made by P for the inputs {1, 2, 3, 4, 5} and {4, 1, 5, 3, 2} respectively. Which one of the following holds?

A t1 = 5

B t1 < t2

C t1 > t2

D t1 = t2

Discuss it

Question 27

You have an array of n elements. Suppose you implement quicksort by always choosing the central element of the array as the pivot. Then the tightest upper bound for the worst case performance is

A O(n^2)

B O(nLogn)

C Theta(nLogn)

D O(n^3)

Discuss it

There are 27 questions to complete.

1 Tweet 0 0