ALGORITHMS

SORTING

(Solutions)

1. Assume that a merge sort algorithm in the worst case takes 30 seconds for an input of size 64. Which of the following most closely approximates the maximum input size of a problem that can be solved in 6 minutes?

A. 256
B. 512
C. 1024
D. 2048

Answer: (B)

Explanation:
Time complexity of merge sort is Θ(n Log n)
c*64 Log64 is 30
c*64*6 is 30
c is 5/64

For time 6 minutes:
5/64*n Log n = 6*60
n Log n = 72*64 = 512 * 9
n = 512.

2. Which one of the following is the recurrence equation for the worst case time complexity of the Quick sort algorithm for sorting n($\geq$ 2) numbers? In the recurrence equations given in the options below, c is a constant.

A. T(n) = 2T (n/2) + cn
B. T(n) = T(n – 1) + T(0) + cn
C. T(n) = 2T (n – 2) + cn
D. T(n) = T(n/2) + cn

Answer: (B)

Explanation:
In worst case, the chosen pivot is always placed at a corner position and recursive call is made for following.

a) for subarray on left of pivot which is of size n-1 in worst case.
b) for subarray on right of pivot which is of size 0 in worst case.

3. Which of the following changes to typical QuickSort improves its performance on average and are generally done in practice?

1) Randomly picking up to make worst case less likely to occur.
2) Calling insertion sort for small sized arrays to reduce recursive calls.
3) QuickSort is tail recursive, so tail call optimizations can be done.
4) A linear time median searching algorithm is used to pick the median, so that the worst case time reduces to O(n Log n)

A. 1 and 2
B. 2, 3, and 4
C. 1, 2 and 3
D. 2, 3 and 4

Answer: (C)

Explanation:
The 4th optimization is generally not used, it reduces the worst case time complexity to O (nLogn), but the hidden constants are very high.

4. Randomized quicksort is an extension of quicksort where the pivot is chosen randomly. What is the worst case complexity of sorting n numbers using randomized quicksort?

A. O(n)
B. O(n log n)
C. O(n^2)
D. O(n!)

Answer: (C)

Explanation:
Randomized quicksort has expected time complexity as O(n Log n), but worst case time complexity remains same. In worst case the randomized function can pick the index of corner element every time.

5. In a permutation a_1…..a_n of n distinct integers, an inversion is a pair (a_i, a_j) such that i < j and $a_i > a_j$. What would be the worst case time complexity of the Insertion Sort algorithm, if the inputs are restricted to permutations of 1…..n with at most n inversions?

A. Θ (n^2)
B. Θ (n log n)
C. Θ ($n^{1.5}$)
D. Θ (n)

Answer: (D)

Explanation:
Insertion sort runs in Θ(n + f(n)) time, where f(n) denotes the number of inversion initially present in the array being sorted.

6. 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. $O(n^3)$

Answer: (A)

Explanation:
The central element may always be an extreme element, therefore time complexity in worst case becomes O(n2).

7. Let P be a Quicksort Program to sort numbers in ascending order using the first element as pivot. Let t_1 and t_2 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. $t_1 = 5$ B. $t_1 < t_2$
C. $t_1 > t_2$ D. $t_1 = t_2$

Answer: (C)

Explanation:
When first element or last element is chosen as pivot, Quick Sort's worst case occurs for the sorted arrays.

In every step of quick sort, numbers are divided as per the following recurrence:
T(n) = T(n-1) + O(n)

8. 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

Answer: (B)

Explanation:
For the case where n/5 elements are in one subset, T(n/5) comparisons are needed for the first subset with n/5 elements, T(4n/5) is for the rest 4n/5 elements, and n is for finding the pivot.
If there are more than n/5 elements in one set then other set will have less than 4n/5 elements and time complexity will be less than T(n/5) + T(4n/5) + n because recursion tree will be more balanced.

9. 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) theta(n)
(B) theta(nLog n)
(C) theta(n^2)
(D) theta(n^2 log n)

A. A
B. B
C. C
D. D

Answer: (B)

Explanation: Answer (B)
The recursion expression becomes:
T(n) = T(n/4) + T(3n/4) + cn

After solving the above recursion, we get theta(n Log n).

10. 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

Answer: (B)

11. 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

Answer: (D)

12. If the number of records to be sorted is small, then ...... sorting can be efficient.

A. Merge

B. Heap

C. Selection

D. Bubble

Answer: (C)

13. The complexity of sorting algorithm measures the …....... as a function of the number n of items to be sorted.

A. average time
B. running time
C. average-case complexity
D. case-complexity

Answer: (B)

14. State True or False for internal sorting algorithms:

(i) Internal sorting are applied when the entire collection if data to be sorted is small enough that the sorting can take place within main memory.
(ii) The time required to read or write is considered to be significant in evaluating the performance of internal sorting.

A. (i)- True, (ii)- True
B. (i)- True, (ii)- False
C. (i)- False, (ii)- True
D. (i)- False, (ii)- False

Answer: (B)

15. .... is putting an element in the appropriate place in a sorted list yields a larger sorted order list.

A. Insertion
B. Extraction
C. Selection
D. Distribution

Answer: (A)

16. Which of the following sorting algorithm is of priority queue sorting type?

A. Bubble sort
B. Insertion sort
C. Merge sort
D. Selection sort

Answer: (D)

17. Partition and exchange sort is ....

A. quick sort
B. tree sort
C. heap sort
D. bubble sort

Answer: (A)

18. Which of the following is an external sorting?

A. Insertion sort
B. Bubble sort
C. Merge sort
D. Tree sort

Answer: (C)

19. Very slow way of sorting is—

A. Insertion sort
B. Heap sort
C. Bubble sort
D. Quick sort

Answer: (A)