

Assignment -1

DAA(KCS503)

1. What is the importance of average case analysis of algorithms? Explain.
2. Find Θ notation for the function $f(n)=27n^2+16n+25$.
3. Let $f(n)$ and $g(n)$ be asymptotic positive functions. Prove or disprove the following conjectures:
 - a. $f(n)=O(g(n))$ implies $g(n)=O(f(n))$
 - b. $f(n)+g(n)=\Theta(\min(f(n),g(n)))$
4. Can the Master method be applied to solve recurrence $T(n)=4T(n/2)+n^2\log n$? Why or why not?
5. Solve the following recurrence relation by master theorem
 - a. $T(n) = 4T(n/2) + n^2 \sqrt{n}$
 - b. $T(n) = T(9n/10) + n$
6. Solve the following recurrence relation by any method known
 - a. $T(n) = T(\sqrt{n}) + 1$
 - b. $T(n) = T(n/3) + T(2n/3) + O(n)$
7. Prove that upper bound of $T(n)=2T(n/2) + n$ is $n\log n$ by Substitution Method.
8. State true or false? The solution to the recurrence $T(n)= 100T(n/99)+\log(n!)$ is $T(n)=\Theta(n\log n)$.
9. Discuss insertion sort, achieve Shell sort using insertion sort and calculate its complexity.
10. What do you mean by stable sort? Name two stable sort algorithms.
11. Show that quick sort's best case running time is $\Omega(n\lg n)$.
12. How do you modify Quick sort algorithm to search an item in a list of elements.
13. Show that there are at most $\lceil n/2^{h+1} \rceil$ nodes of height h in any n - element heap.
14. What is the effect of calling Max-Heapify (A,i) when the element $A[i]$ is larger than its children?
15. Illustrate the functioning of Heap Sort on the following array- $A=$
25,57,48,37,12,92,86,33
16. What are the minimum and maximum numbers of elements in a heap of height h ?

17. Why we call Heapify procedure only on first half elements of the given array while building a heap? Explore the utility of heapsize attribute of the array in these algorithms.
18. Prove that any comparison sort algorithm require $\Omega(n \log n)$ comparisons in worst case.
19. Write the counting sort algorithm and discuss the complexity. List the limitation of the algorithm.

The array A is given below apply counting sort and so each steps of the algorithm on the data.

A = < 0, 1, 5, 3, 2, 1, 0 >

20. Prove that Counting Sort is stable sort.