Assignment 1 (DAA)

Last Date of Submission- 23/01/2025

1. State True / False with explanation
   a. log n = Ω (n log n)
   b. $2^n = \Omega (2^{n/2})$
   c. $n^2 + n = \Omega (n^3)$
   d. $n! = \Omega(n^n)$
   e. log n!= o (n log n)
   f. $x^{n+1} = o (x^n)$
   g. $n^2 (\log n)^{10} = o (n^{2+0.1})$
   h. If given f(n) = O(g (n)) then $2^{f(n)} = O (2^{g(n)})$ (T/F)
   i. $f(n) = O (f(n^2))$
   j. $f(n) = n^{\sqrt{n}}$ and $g(n) = (\sqrt{n})^{\log n}$ then f(n) asymptotic bigger than g(n)
   k. $f(n) = n^{1 + \sin n}$, g(n)=n then f(n) and g(n) are non-comparable
2. Write the function in asymptotic increasing order.
   a. $X^n, X^{2n}, X^{n+1}, 2^{n/2}, 2^n, \sqrt{n}^n, n^{\log \sqrt{n}}$
   b. $n^{\log n}, (\log n)^{\log n}, (\log n)!, 2^n, (\sqrt{n})^{\log n}, (\log n)^n$
   c. $n^{\sqrt{n}}, n^{\log n}$, n log n, log n!, (log n)!
   d. f1 (n)= log log log n^k, f2(n)= log $\log^k$ log n, f3(n) = $\log^k$ log log n, f4(n) = log log $\log^k$ n
      Write the asymptotic order of f1, f2, f3 and f4.
3. Solve the following recurrences using Master method.
   a. $T(n)=3T(n/2) + n^2$
   b. $T(n)=2T(n/2) + \sqrt{n}$
   c. $T(n)= T(n/2) + \sqrt{n}$
   d. T(n)= T(n/2) + 1
   e. $T(n)= 4T(n/2)+ n^2 (\log n)^2$
4. Discuss 2-way merge sort algorithm with divide and conquer technique.
5. Consider the following recursive algorithm for computing the sum of the first n cubes:

$$S(n) = 1^3 + 2^3 + \cdots + n^3.$$

ALGORITHM $S(n)$

```
//Input: A positive integer n
//Output: The sum of the first n cubes
if n = 1 return 1
else return S(n − 1) + n ∗ n ∗ n
```

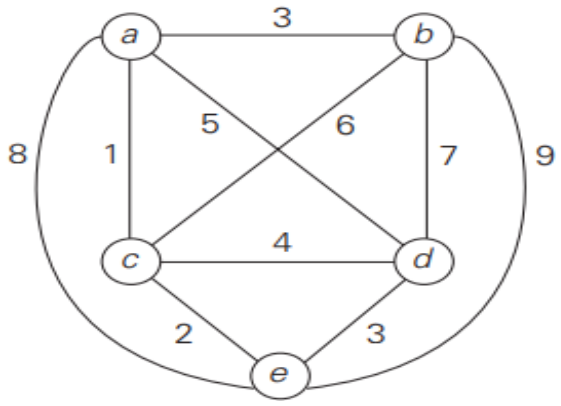
a. Set up and solve a recurrence relation for the number of times the algorithm's basic operation is executed.

b. How does this algorithm compare with the straightforward nonrecursive algorithm for computing this sum?

6. Compare and discuss the recurrence relation, relative time (Best, Worst case) and space complexity of Merge, Quick, and Binary search.
7. Solve the following assignment problem using brute force technique

		Machines				
		A	**B**	**C**	**D**	**E**
Job	**1**	5	7	11	6	7
	2	8	5	5	6	5
	3	6	7	10	7	3
	4	10	4	8	2	4

8. Apply quicksort to sort the list C, E, N, T, R, U, M, K, I, D, S in alphabetical order. Draw the tree of the recursive calls made using divide and conquer algorithm design technique.
9. Write recursive algorithm for binary search and analyze its time complexity using Masters theorem.
10. Solve the following instance of the knapsack problem with knapsack capacity 8 by the brute force algorithm: <I1, I2, I3, I4>, <w1=2, w2=4, w3=3, w4=5>, <v1=14, v2=4, v3=15, v4=20>
11. Solve the following travelling salesman problem using Brute force technique.



12. (a) What is the largest number of key comparisons made by binary search in searching for a key in the following array?

3	14	27	31	39	42	55	70	74	81	85	93	98

(b) List all the keys of this array that will require the largest number of key comparisons when searched for by binary search.

(c) Find the average number of key comparisons made by binary search in a successful search in this array. Assume that each key is searched for with the same probability.

(d) Find the average number of key comparisons made by binary search in an unsuccessful search in this array. Assume that searches for keys in each of the 14 intervals formed by the array's elements are equally likely.