1.	What is an algorithm? Explain various properties of an algorithm.
2.	What is an algorithm? How it differs from flowchart?
3.	Define Algorithm, Time Complexity and Space Complexity.
4.	Define Algorithm. Discuss key characteristics of algorithm.
5.	(i)Explain why analysis of algorithms is important? Explain: Worst Case, Best Case & Average Case Complexity. (ii) Define: Optimal Solution, Feasible solution, Principle of Optimality.
6.	Arrange following rate of growth in increasing order. $2^n, n \log \log n, n^2, 1, n, \log \log n, n!, n^3$
9.	Explain Asymptotic notation. Arrange the growth rate of in increasing order of growth. $2^n, n^2, 1, \log \log n, n \log \log n, 3^n, n$
10.	What do you mean by Asymptotic Notations? Explain.
11.	Solve the recurrence $T(n) = 7 * T(n) + n^3$
12.	Explain counting sort with the help of an example and code. Also, how is counting sort different from other sorting techniques like bubble sort or inserting sort or others.
13.	Explain radix sort with the help of an example and code. Also explain the complexity of radix sort.
14.	Discuss the time and space complexity analysis of Fibonacci sequence with the help of an example. Can this be improved by using dynamic programming? Explain the answer.
15.	Which are the basic steps of counting sort? Write counting sort algorithm. Derive its time complexity in worst case.
16.	Solve following recurrence relation using suitable method and express your answer using Big-oh (O) notation. $T(n) = 2 * T(n/2) + n^2$