PRANVEER SINGH INSTITUTE OF TECHNOLOGY KANPUR

Odd Semester Session 2023-24 CT -1

B. Tech 5^{th} Semester

Design & Analysis of Algorithms (KCS-503)

CO's	Course Outcome
CO1	Define [L1: Remember] Algorithms, Asymptotic Notations, Recurrences, Advanced D Structures, Algorithm design approaches and complexity classes.
CO2	Explain [L2: Understand] Various Algorithm design approaches and advanced data structures.
CO3	Apply [L3: Apply] appropriate algorithm design approach to solve a computational proble Ability to apply a suitable recurrence solving method to solve a recurrence.
CO4	Analyse [L4: Analyze] an Algorithm to solve a computational problem.

Time: 1.5 Hrs. M. M. 15

Section A

Q1. Attempt all questions: *(1X3 = 3 Marks)*

a) Define little oh (o) and little omega (ω) with function of Limit. CO1

b) State the Substitution method show that CO1

$$T(n) = 2T\left(\frac{n}{2}\right) + n \in O(n\log(n))$$

c) Interpret the following three functions arrange the functions in the increasing order of *asymptotic growth rate?* CO2

$$f_1 = 10^n \quad f_2 = n^{\log n} \quad f_3 = n^{\sqrt{n}}$$

Section B

Q2. Attempt all questions: **(2X4 = 8 Marks)**

a i) Explain insertion sort (Iterative) algorithm with example and prove that if array is sorted the time complexity will be O (n) else it will be O (n^2). CO2

Or

ii) Explain Bubble Sort (Recursive) algorithm and its recurrence relation and prove the complexity is O (n^2). CO2

b i) Solve the following recurrence using recursion tree method. CO4

$$T(n) = \begin{cases} 1 & n = 1 \\ T\left(\frac{n}{10}\right) + T\left(\frac{9n}{10}\right) + cn & n > 1 \end{cases}$$

Or

ii) Solve the following recurrence using recursion tree method and describe the recurrence relation. CO4

$$T(n) = \begin{cases} 1 & n = 1 \\ T\left(\frac{n}{3}\right) + T\left(\frac{2n}{3}\right) + cn & n > 1 \end{cases}$$

c i) Explain Linear Search (Recursive) describe the recurrence relation and prove the time Complexity of both recursive and iterative are same. CO2

Or

ii) Demonstrate Recursive Selection sort in the following given array A [6, 14, 3, 25, 2, 10, 20, 7, 6] and also derived the complexity from recurrence relation. CO2

d i) Calculate the complexity if running time of an algorithm is represented by the following recurrence relation. CO3

$$T(n) = \begin{cases} 1 & n = 1 \\ 2T\left(\frac{n}{2}\right) + n.\log n & n > 1. \end{cases}$$

Or

ii) Calculate the complexity of following recurrence relation by using substitution method CO3

$$T(n) = \begin{cases} 2 & 0 < n \leq 2 \\ 2T(\sqrt{n}) + \log n & n > 2 \end{cases}$$

Section C

(4X1 = 4 Marks)

Q3 Attempt all questions:

i) Calculate the smallest value of *a* such that *B* is asymptotically faster than *A*? The recurrence T (n) = 7T(n/3) + n^2 describes the running time of an algorithm *A*. Another competing algorithm *B* has a running time of S(n) = a S(n/9) + n^2. CO3

Or

ii) Illustrate the Recursive Insertion sort on the array $A = \{6, 14, 3, 25, 2, 10, 20, 7, 6\}$ and also derived the recurrence relation and its complexity. CO3