SSN COLLEGE OF ENGINEERING, KALAVAKKAM – 603 110
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
B.E. Computer Science and Engineering
CS8451 Design and Analysis of Algorithms

Date: 21.01.2019, 8.00-9.30 AM **UNIT TEST – 1** **Max. Marks: 50**
Academic Year: 2018-2019 Even **Batch: 2017-2021**
Semester: 4 **Faculty: V. Balasubramanian / S. Manisha**

Qn. No	Part – A (5 * 2 = 10)	Marks	(KL,COn)
1	**ALGORITHM** *Riddle*$(A[0..n-1])$ //Input: An array $A[0..n-1]$ of real numbers **if** $n = 1$ **return** $A[0]$ **else** *temp* $\leftarrow$ *Riddle*$(A[0..n-2])$ **if** *temp* $\leq A[n-1]$ **return** *temp* **else return** $A[n-1]$ What does this algorithm does?	2	K2,CO1
2	Find the time complexity? for i $\leftarrow$ 1 to m for j $\leftarrow$ 1 to n c[i , j] $\leftarrow$ a[i , j] + b[i , j] end end	2	K3,CO1
3	Find the order of growth of the sum $\sum_{i=0}^{n-1}(i^2+1)^2$	2	K2,CO1
4	What is the smallest value of n such that an algorithm whose running time is $100n^2$ runs faster than an algorithm whose running time is 2^n on the same machine?	2	K2,CO1
5	Design a brute-force algorithm for computing the value of a polynomial at a given point x_0 $p(x) = a_n x^n + a_{n-1}x^{n-1} + \cdots + a_1 x + a_0$ and determine its worst-case efficiency class.	2	K4,CO2

Part – B Answer all questions (13+13)

8	a)	Consider the problem of finding the smallest and largest element in an array of *n* numbers.		
		(i) Design a pre-sorting based algorithm for solving this problem and determine its efficiency.	8	K3,CO1
		(ii) Compare the efficiency of pre-sorting algorithm with brute force algorithm	5	
		OR		
9	a)	Define Big-oh notation, Big-Ω and Big-Θ notation. Depict the same graphically and explain	13	K2,CO1
10	a)	Explain the efficiency of Tower of Hanoi puzzle using recursion.	7	K2,CO1
	b)	Find the number of times the basic operation happens in	6	K2,CO1

matrix multiplication.

OR

11	a) Suppose we are comparing implementations of insertion sort and Merge sort on the same machine. For inputs of size n, insertion sort runs in $8n^2$ steps, while merge sort runs in 64nlogn steps (Recall that Log is the log base 2 function). For which values of n does Insertion sort beat merge sort?	6	K3,CO1
	b) Consider two algorithms A and B for solving the same problem running on two machines 1 and 2. Machine 1 executes 10^9 (1 billion) instructions per second, and machine 2 executes 10^7 (10 million) instructions per second. Algorithm A requires $2n^2$ instructions and runs on machine 1; algorithm B requires $50n\log_{10}n$ instructions and runs on machine 2. (a) Calculate the running time of the two algorithms for inputs of sizes 100, 1000, 10000. (b) Which is better— algorithm A on machine 1, or algorithm B on machine 2? Why?	7	K3,CO1

Part – C (14)

12	a) (i) If $t(n) = \frac{1}{2}n^2$, then what is t(2n) ? (ii) Solve the recurrence relation: x(n) = x(n-1) +5 for n>1 , x(1) = 0. (iii). Arrange these functions in increasing order of asymptotic growth: cn, nlogn, n^2, logn, n, n!, n^3.	6	K2,CO1
	b) Show how to implement stack using two queues. Analyse the running time of the stack operations.	8	K4,CO1

OR

13	a) If you have to solve the searching problem for a list of n numbers, how can you take advantage of the fact that the list is known to be sorted? Give separate answers for (i) list represented as arrays (ii) list represented as linked lists. Compare the time complexities involved in the analysis of both the algorithms.	6	K4,CO2
	b) What is closest pair problem, convex hull problem?	4+4	K2,CO2
	c) Find an optimal solution to the knapsack instance n=4, W=10, {p1…p4}={10,40,30,50}, {w1…w4} = {5,4,6,3}.		

************B E S T O F L U C K************

Prepared by	

Reviewed by HoD, CSE