SSN COLLEGE OF ENGINEERING, KALAVAKKAM – 603 110
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
B.E. Computer Science and Engineering
CS6402   Design and Analysis of Algorithms
Date: 11.02.2017, 8.00-9.30 AM   UNIT TEST – 1 (Retest)   Max. Marks: 50
Academic Year: 2016-2017 Even   Batch: 2015-2019
Semester: 4   Faculty: Dr. R. S. Milton / Mr. V. Balasubramanian

Qn. No	Part – A (5 * 2 = 10)	Marks	(KL,COn)
1	Write the recursive Fibonaaci algorithm and its recurrence relation	2	K1,CO1

2	**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	K1,CO1

3	Find the time complexity?
	  for i ← 1 to m
	    for j ← 1 to n
	      c[i , j] ← a[i , j] + b[i , j]
	    end
	  end
		2	K3,CO3

4	Solve the recurrence relation $x(n) = x(n-1) + n$ for n>0 , x(0) = 0	2	K1,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) 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.	8	K3,CO3
	b) Write and solve the recurrence relation for computing factorial of a number.	5	K1,CO1

OR

9		8	K3,CO3

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)   Analyse the

algorithm and find what this algorithm computes, and solve the recurrence relation for the basic operation.

	b) Solve the recurrence relation x(n) = 2x(n/2) + n for n>1 , x(1) = 1	5	K1,CO1
10	a) (i) If t(n) = ½ n2, then what is t(2n) ?	6	K1,CO1
	(iii) Solve the recurrence relation: x(n) = x(n-1) +5 for n>1 , x(1) = 0.		
	(iv). Arrange these functions in increasing order of asymptotic growth: cn, nlogn, n2, logn, n, n!, n3.		
	b) Show how to implement stack using two queues. Analyse the running time of the stack operations.	7	K3,CO3

OR

11	a) Solve the recurrence relation x(n) = x(n/3) + 1 for n>1 , x(1) = 1	5	K1,CO1
	b) Write the recursive and iterative algorithm for computing nth fibonacci number and solve the recurrence relation.	8	K4,CO2

Part – C (14)

12	a) Explain the efficiency of Tower of Hanoi puzzle using recursion.	8	K3,CO3
	b) What is closest pair problem, convex hull problem?	6	K1,CO1

OR

13	a) For the one-dimensional version of the closest-pair problem, i.e., for the problem of finding two closest numbers among a given set of n real numbers, design an algorithm that is directly based on the divide-and-conquer technique and determine its efficiency class.	4	K3,CO3

ALGORITHM $Q(n)$
```
//Input: A positive integer n
if n = 1 return 1
else return Q(n − 1) + 2 ∗ n − 1
```

		10	K3,CO3

b)   Set up a recurrence relation for this function's values and solve it to determine what this algorithm computes?

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

Prepared by	Reviewed by HoD, CSE

SSN