

Sri Sivasubramaniya Nadar College of Engineering, Kalavakkam – 603 110

(An Autonomous Institution, Affiliated to Anna University, Chennai)

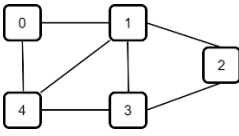
Department of Computer Science and Engineering

Continuous Assessment Test – II

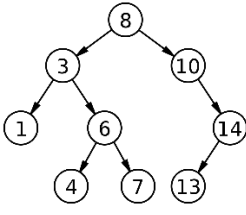
Question Paper

Degree & Branch	BE (CSE)				Semester	III
Subject Code & Name	UCS1302 Data Structures				Regulation:	2018
Academic Year	2021-2021 ODD	Batch	2020-2024	Date	01.12.2021	FN
Time: 90 Minutes	Answer All Questions				Maximum: 50 Marks	

Part – A (6×2 = 12 Marks)

<KL2>	1. Show the expression tree for the arithmetic expression $a*b/(c+d/e)$	<CO3>
<KL1>	2. What is the maximum number of nodes in a binary tree of height n.	<CO3>
<KL2>	3. Outline the recursive implementation of a function that returns the smallest element in the binary search tree	<CO3>
<KL1>	4. Define the properties of AVL tree. Give an example of an AVL tree.	<CO3>
<KL2>	5. Outline the algorithm for printing the degree of each vertex in a given undirected graph G represented by adjacency matrix.	<CO4>
<KL3>	6. Apply BFS to the following graph and write the nodes visited, starting from node 0. 	<CO4>

Part – B (3×6 = 18 Marks)

<KL3>	7. Develop a C function for in order traversal of a binary tree. Trace the function, for the following binary tree. 	<CO3>
<KL2>	8. Outline the algorithm for searching an element in a Binary search tree.	<CO3>
<KL3>	9. Given the following array of integers, apply selection sort and write the contents of the array after FOUR passes of the outermost loop. Write the contents of the array after ONE pass of shell sort, using interval = 3. $A[] = \{22, 11, 34, -5, 3, 40, 9, 16, 6\}$	<CO5>

Part – C (2×10 = 20 Marks)

<KL3>	<p>10. Given input items 25,54,46,28,19,45,27,48,75. Apply the following operations showing the tree diagram for each step.</p> <p>a) Construct BST. Delete 46</p> <p>b) Construct AVL tree. Insert 13</p>	<CO3>																									
(OR)																											
<KL3>	<p>11. Develop the ADT for PriorityQ using a MinHeap and the algorithm for DELMIN(Q). Trace your algorithm for the following heap. Clearly show the array as well as the tree representations of the heap during trace.</p> <div style="text-align: center;"> <p>17</p> <p>20 26</p> <p>24 31 27 30</p> <p>49 52</p> </div>	<CO3>																									
<KL2>	<p>12. Outline the algorithm for $dfs(G)$ that visits all the vertices of a directed graph G. Trace your algorithm for a directed graph represented as an adjacency matrix given below.</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td></td> <td>A</td> <td>B</td> <td>C</td> <td>D</td> </tr> <tr> <td>A</td> <td>0</td> <td>1</td> <td>1</td> <td>0</td> </tr> <tr> <td>B</td> <td>0</td> <td>0</td> <td>1</td> <td>1</td> </tr> <tr> <td>C</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> </tr> <tr> <td>D</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> </tr> </table>		A	B	C	D	A	0	1	1	0	B	0	0	1	1	C	0	0	0	0	D	0	0	0	0	<CO4>
	A	B	C	D																							
A	0	1	1	0																							
B	0	0	1	1																							
C	0	0	0	0																							
D	0	0	0	0																							
(OR)																											
<KL2>	<p>13. Given an array of names of the cities sorted in lexical order. Outline the algorithm or C function that searches for a city in the array. Compute the time complexity.</p>	<CO4>																									
