U.S.N.										

B.M.S. College of Engineering, Bengaluru-560019

Autonomous Institute Affiliated to VTU

January / February 2025 Semester End Main Examinations

Programme: B.E. **Semester: V**

Branch: Computer Science and Engineering **Duration: 3 hrs.**

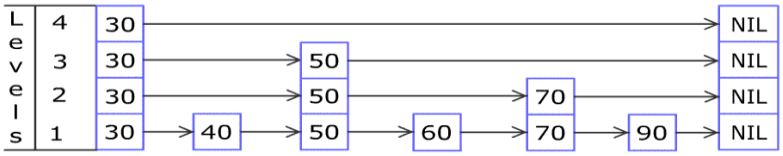
Course Code: 20CS5PEADS **Max Marks: 100**

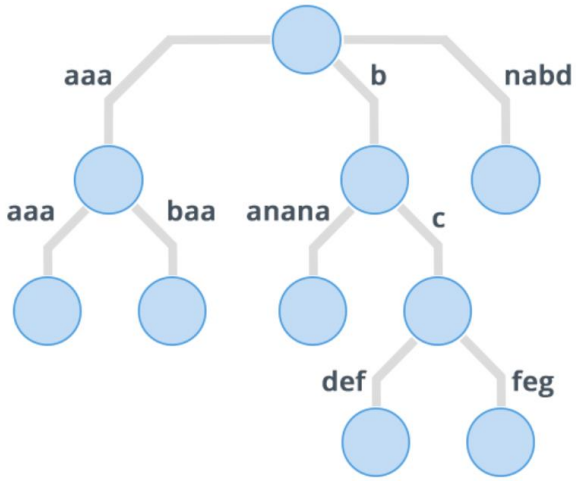
Course: Advanced Data Structures

Instructions: 1. Answer any FIVE full questions, choosing one full question from each unit.
2. Missing data, if any, may be suitably assumed.

Important Note: Completing your answers, compulsorily draw diagonal cross lines on the remaining blank pages. Revealing of identification, appeal to evaluator will be treated as malpractice.

		UNIT - I	*CO*	*PO*	Marks
1	a)	What is an unrolled linked list? Explain with an example.	*1*	*2*	**4**
	b)	Consider the following self-organizing list implemented using count method. Input list: 1, 2, 3, 4, 5 Value searched: 4 Output list: 4, 1, 2, 3, 5 Modified input list now: 4, 1, 2, 3, 5 Value searched: 5 Value searched: 5 Value searched: 2 Analyze the input given and draw the output list and justify your answer.	*1*	*2*	**6**
	c)	Write a program to implement the following operations on a memory efficient linked list: (a) A function to insert a new node at the beginning of the list (b) A function to traverse the list in forward direction	*2*	*3*	**10**
		OR			
2	a)	Compare move to front method and transpose method for self-organizing list and analyze which method gives better results.	*1*	*2*	**6**
	b)	Demonstrate with an example the concept of generic list.	*1*	*2*	**6**
	c)	Analyze the skip list diagram given below and delete the node with value 70. Show the updated skip list with appropriate pointers and levels.	*1*	*2*	**8**

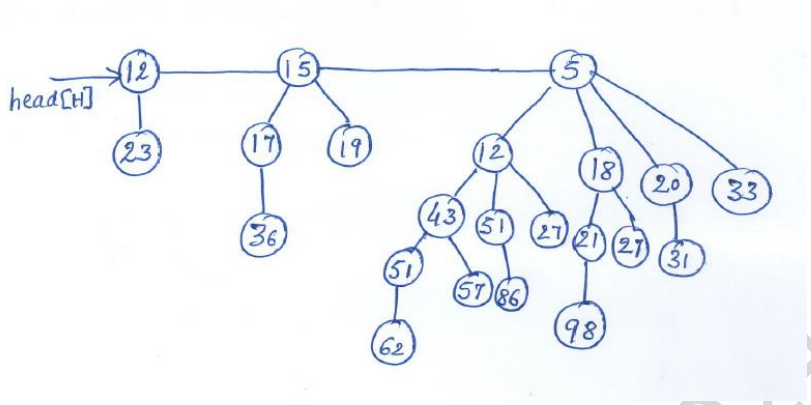
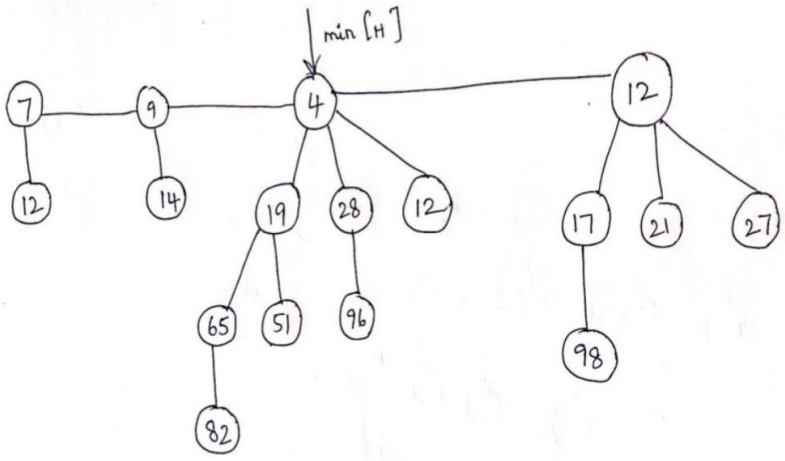
						
			UNIT - II			
3	a)		Construct AVL tree by inserting the following elements successively from an empty tree 100, 200, 300, 250, 270, 70, 40.	*1*	*2*	**10**
	b)		Construct a Red Black Tree for the following numbers: 9, 15, 7, 89, 91, 23, 51, 22 Show all the steps clearly.	*1*	*2*	**10**
			OR			
4	a)		Demonstrate the two types of rotations used in splay tree with an example for each.	*1*	*2*	**5**
	b)		Create a B-tree of order 5 for the following: 8, 9, 10, 11, 15, 16, 17, 18, 20, 23	*1*	*2*	**10**
	c)		Analyze with an example how AVL tree searching operation works faster than normal Binary Search Tree.	*1*	*2*	**5**
			UNIT - III			
5	a)		Construct Trie and suffix array tree for the following strings: “mango”, “mat”, “man”, “an”, “ant”, “antelope”, “bake”, “bad” Show which data structure is better in terms of time complexity for searching.	*1*	*2*	**5**
	b)		Construct a Segment Tree for the following values: {7, -5, 3, 10, -2, 11, 9, 12, -4, 13} and show the maximum of (3,6) is computed using the Segment tree. Show the paths used for computation clearly.	*1*	*2*	**10**
	c)		Write the procedure to compute the parent node in a Fenwick tree with an example.	*1*	*2*	**5**
			OR			
6	a)		Construct a 2-dimensional tree for the following points: (5,12), (7,13), (1,9), (8,11), (12,5), (10,3)	*1*	*2*	**5**
	b)		Construct a Fenwick Tree for the following array {3,-1,2,1,4,2,8,7,-2,5,11,3} Also show how the sum from 0 to 6 is computed and searching happens in the tree. Clearly show how the parent is computed for each node and how the tree structure is formed.	*1*	*2*	**10**
	c)		Analyze the Suffix Tree given below:	*1*	*2*	**5**



Add the string aaabc and bcdff to the above tree and draw the updated tree structure.
Also justify how Suffix Tree performs searching faster than Tries.

			UNIT - IV			
	7	a)	Construct a Hash table for the following numbers: 8, 17, 19, 54, 32, 21, 71, 89, 39, 26 Show how collision is resolved using the following techniques: (1) Linear Probing (2) Quadratic Probing (3) Double Hashing	*3*	*1*	**10**
		b)	Construct a Hash table for the following numbers: 9, 14, 18, 76, 27, 65, 34, 51, 46, 83, 94, 44, 112, 71, 31 Show how collision is resolved using Extendible Hashing with each step demonstrated clearly. Also, mention how directory expansion and bucket splitting happen. Bucket limit=3.	*3*	*1*	**10**
			OR			
	8	a)	Analyse the hash table given below and determine for which values collision occurs. And also determine how the hash function is used to resolve collision. Assume quadratic probing is used for collision resolution.	*3*	*1*	**5**
		b)	Construct a Hash table for the following numbers: 18, 21, 45, 67, 11, 5, 61, 23, 77, 99, 19, 56, 61, 29, 68 Show how collision is resolved using Extendible Hashing with each step demonstrated clearly. Also, mention how directory expansion and bucket splitting happens. Bucket limit=3.	*3*	*1*	**10**
		c)	Explain rehashing with an example.	*3*	*1*	**5**

0	**1**	**2**	**3**	**4**	**5**	**6**
14	**21**	**47**	**38**		**40**	**13**

			UNIT - V			
	9	a)	Construct a Binomial Heap with 3 Binomial trees of order 0,3,4 and demonstrate how minimum node is extracted with all the steps clearly.	*1*	*2*	**10**
		b)	Analyze the Binomial Heap given below and demonstrate decreasing the value 62 to 6. 	*1*	*2*	**5**
		c)	What is a Binomial Tree? Explain the properties of Binomial trees.	*1*	*2*	**5**
			OR			
	10	a)	Explain the structure of a Fibonacci Heap with an example.	*1*	*2*	**5**
		b)	Analyze the following code and complete the function definition for BINOMIAL-HEAP-UNION: **BINOMIAL-HEAP-INSERT(H,*x*)** H' <-- MAKE-BINOMIAL-HEAP() p[x] <-- NIL child[x] <-- NIL sibling[x] <-- NIL degree[x] <-- 0 head[H'] <-- x **H <-- BINOMIAL-HEAP-UNION(H,H')**	*1*	*2*	**5**
		c)	For the below Fibonacci Heap, show how the value 65 and 98 are decreased to values 23 and 13 respectively. Node 18 is the marked node. Show all the steps clearly. 	*1*	*2*	**10**

******