

**PRAGATI ENGINEERING COLLEGE: SURAMPALEM**  
(AUTONOMOUS)

**II B.Tech I Semester Supplementary Examinations, June - 2024**

**ADVANCED DATA STRUCTURES THROUGH C**  
(Common to CSE, CSE(AI&ML), CSE(DS), CSE(AI) & IT)

Time: 3 hours

Max. Marks: 70

**Answer ONE Question from each Unit**  
**All Questions Carry Equal Marks**

Q. No.	Questions	BTL	CO	Marks
<b>UNIT – I</b>				
1.	a) Explain the concept of static hashing and how hash tables are utilized in this approach.	K2	CO1	7M
	b) Illustrate the importance of using secure hash functions in designing hash tables.	K3	CO1	7M
<b>OR</b>				
2.	a) Analyze the significance of hash functions in static hashing.	K2	CO1	7M
	b) Contrast directory-less dynamic hashing and dynamic hashing using directories, by providing examples where directory-less dynamic hashing might be preferable.	K3	CO1	7M
<b>UNIT – II</b>				
3.	a) Illustrate the heap operations decrease key, delete, and heapify. Explain their significance in maintaining the integrity of a heap and how they contribute to efficient heap-based algorithms	K2	CO2	7M
	b) Choose two specific applications, such as the Selection Problem and Event Simulation, and elaborate on how priority queues are employed to enhance algorithmic efficiency in these scenarios.	K3	CO2	7M
<b>OR</b>				
4.	a) Illustrate the heap operations insertion and extraction in the context of heap data structure.	K3	CO2	7M
	b) Provide brief implementation detail of binomial queue operations.	K2	CO2	7M
<b>UNIT – III</b>				
5.	a) Provide a step-by-step illustration of the dynamic programming approach for constructing optimal binary search trees.	K2	CO3	7M
	b) Illustrate how Red-Black Trees are self-balancing and the advantages they offer in terms of efficient searching.	K2	CO3	7M
<b>OR</b>				
6.	a) Provide a detailed explanation of how AVL trees ensure logarithmic height and analyze the time complexity of fundamental operations.	K3	CO3	7M

	b)	Write the algorithm for deleting a node from a Red-Black Tree. Analyze how the tree maintains balance during this delete operation in a sample Red-Black Tree.	K3	CO3	7M
<b>UNIT – IV</b>					
7.	a)	Define M-Way Search Trees and highlight their key properties. Discuss the advantages and disadvantages of M-Way Search Trees in comparison to binary search trees.	K3	CO4	7M
	b)	Define B+-Trees and discuss their key characteristics. Explain the searching algorithm in a B+-Tree and how it differs from B-Trees. Highlight the advantages of using B+-Trees in certain scenarios.	K3	CO4	7M
<b>OR</b>					
8.	a)	Explain the algorithm for searching in an M-Way Search Tree. Discuss the time complexity of the search operation and how the M-Way structure contributes to efficient retrieval.	K3	CO4	7M
	b)	Define B-Trees and discuss their key properties. Explain how B-Trees maintain balance and efficient search operations. Calculate the maximum and minimum number of elements in a B-Tree of a given order.	K3	CO4	7M
<b>UNIT – V</b>					
9.	a)	Define Digital Search Trees and discuss their key characteristics. Explain the operations of search, insert, and delete in the context of Digital Search Trees.	K2	CO5	7M
	b)	Explain the concept of Compressed Binary Tries in detail, highlighting how they optimize space compared to regular binary tries.	K2	CO5	7M
<b>OR</b>					
10.	a)	Contrast Binary Tries and Patricia trees, focusing on their structures and applications.	K3	CO5	7M
	b)	Define the height of a Trie and discuss factors that influence it. Analyze the space requirements of Tries, considering factors such as node size, branching, and compression techniques.	K3	CO5	7M