Regulations: 2022

A

Reg No: 7 3 7 6 2 4 1 C S 3 0 9

BANNARI AMMAN INSTITUTE OF TECHNOLOGY
(An Autonomous Institution Affiliated to Anna University, Chennai)
SATHYAMANGALAM - 638 401

Periodical Test - II - November 2025
III Semester

22AI302 / 22AM302 / 22CS302 / 22IT302 & DATA STRUCTURES I

Degree & Branch: B.E./ B.Tech. & AD, AL, CS, IT

Time: 1:30 hrs

Maximum: 50 Marks

Instructions:
1. Students should not mark/write anything on the Question Paper other than the register number.
2. Section A of the Question Paper contains questions for 15 Marks. Sections B and C contain questions for 30 Marks each.
3. Students can attempt answering any two out of three subsections in each section. The maximum mark is limited to 10 in section A and 20 in section B & C.

Q. No.		Questions	
Section - A		COURSE OUTCOME 3	MAXIMUM: 10 MARKS
A1	(i)	A real-time stock price access system must frequently retrieve and update stock values at specific positions. Linked lists and arrays are possible data structures for storing these prices. Find the key disadvantage of using a linked list over an array in this system: a) Takes more memory b) Reduces memory usage by avoiding fixed size c) Supports static memory allocation d) Allows dynamic memory (1 Mark - [Ap/C, 2])	
	(ii)	Map each real-time scenario with the most appropriate node-based data structure that efficiently supports the required operations. Scenario: 1. Browser navigation with forward and backward movement 2. Music playlist that loops back to the first song after the last 3. Real-time data insertion and fast search, such as in autocomplete 4. Managing tasks in a queue with dynamic memory allocation Data structure: a. Singly Linked List b. Doubly Linked List c. Circular Linked List d. Skip List (4 Marks - [U/C, 2])	
A2	(i)	In a real-time navigation system, route points are stored in a singly linked list to represent the path. Predict the impact of inserting a new location node in the middle of this linked list. a) Requires shifting all elements	

b) Insertion is slow due to index tracking
c) Only pointers are adjusted, insertion is efficient
d) Middle insertions are not possible

(1 Mark - [Ap/C, 2])

(ii) A social media analytics engine uses a multi-level skip list for ranking trending hashtags, enabling real-time insertions, range queries, and concurrent access. Analyze the engineer's claims about the skip list's structure and performance.
Identify the correct statements, and justify each based on skip list principles.
Statements:
1.Skip list supports expected O(log n) insertion and search time in average cases.
2.Each level of the skip list must contain an equal number of elements.
3.Skip lists provide efficient support for executing bounded range queries.
4.Maintaining balance in skip lists requires complex tree rotation logic.
5.Nodes may contain multiple forward pointers depending on their level.
6.Increasing the probability p for node promotion improves worst-case performance.
7.Skip list structures are well-suited for concurrent operations due to their layered design.
8.Duplicate values are stored in the same node to save memory and simplify lookup.

(4 Marks - [Ap/C, 2])

A3 (i) A file explorer uses a doubly linked list to manage folder navigation history. The following code simulates movement from one folder to another:
Code:

```
class Node:
    def __init__(self, data):
        self.data = data
        self.prev = None
        self.next = None
head = Node("Folder1")
second = Node("Folder2")
head.next = second
second.prev = head
print(second.prev.data)
```

Select the correct output and the corresponding navigation action:
a) Folder1 – Back navigation from Folder2 to Folder1
b) Folder2 – Forward navigation from Folder1 to Folder2
c) None – Initial folder without backward link
d) Error – Missing link causes exception

(1 Mark - [U/C, 2])

(ii) Consider the figure named deletion of an element which depicts the process of deleting NODE B from the linked list.
Convert the figure to the equivalent code to perform the operation.

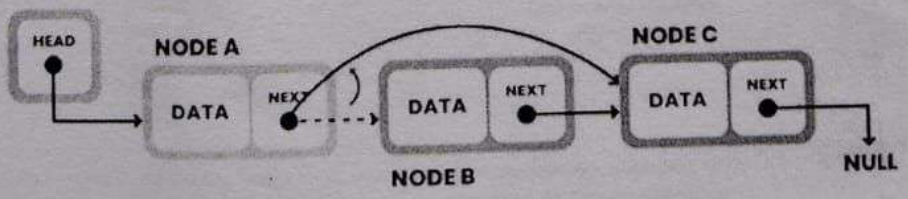

Figure: Deletion of an element

(4 Marks - [An/C, 2])

Section - B	COURSE OUTCOME 4	MAXIMUM: 20 MARKS

B1	(i)	An online dictionary platform uses a hash table to store words and their definitions. The system must efficiently resolve collisions when two words hash to the same index in the table.Select a type of collision resolution technique that is not used in hash table . a) Linear probing b) Separate chaining c) Hashing with rehashing d) Quadratic probing (1 Mark - [U/C, 2])
	(ii)	As a hash table grows and more elements are inserted, it may encounter performance issues due to a high load factor or excessive collisions. In such cases, rehashing becomes necessary to maintain efficient operations.Prefer the scenario in which the rehashing is necessary in a hash table due to high load factor. a)When there are too many keys inserted into the hash table b)When all elements are sorted c)When the hash table has no collisions d)When the hash table is empty (1 Mark - [U/C, 2])
	(iii)	Consider a hash table of size seven, with starting index zero and the hash function "key mod 7" to insert the following sequence of keys(50,700,76,85,92,73 and 101)in the hash table. Select the approriate index bsed on hash function while inserting sequence of keys if collision occurs use separate chaining techniques. Index 0 Index 1 Index 2 Index 3 Index 4 Index 5 Index 6 (2 Marks - [U/P, 2])
	(iv)	An online library needs to assign unique locations in a hash table to store information about books. Each book has a unique ISBN number, which will be used as the key. To store these books in a hash table, different hash functions can be applied to map ISBN numbers to specific indices.The ISBN number to store the book using hash methods given below. The hash value is 1000. Predict the index values you expect to storing data in hash tables using the specified methods. i). Apply Division Method ISBN = 9780131103627 n = 1000 ii). Apply Mid-square Method ISBN = 9780131103627 9780131103627^2 = 9565207421899044249 iii). Apply Folding Method ISBN = 9780131103627 split the number into parts of 4 digits (6 Marks - [An/P, 2])
B2	(i)	A list contains details of multiple books with title, author, and year. Each item should represent one book's full information. Pick the data structure that suits this scenario.

c)Rehashing happens when there are no collisions in a hash table
d)Rehashing decreases the load factor of a hash table

(1 Mark - [U/F, 1])

(ii) In a hash table, two different keys may generate the same hash index. This situation needs to be handled properly to maintain efficient data access. Identify the result when two keys hash to the same index in a hash table
a) Collision
b) Rehashing
c) Load factor increase
d) Clustering

(1 Mark - [U/C, 2])

(iii) A cryptographic hash function comprising a thread of digits generated by using a one - way hashing formula. Message digets are set up to prevent the details of a piese of media or data and to identify changes and modifications to any part of a message. For example, While storing a file in any open cloud space, one would want to predict the integrity of the file. This problem can be solved using the message digest.
You are given the following input values

4322, 1471, 9679, 1989, 6171, 6173, 4199

Use the hash function hash(x)=xmod10

Construct the hash table by storing each input value in the corresponding index based on its hash value. If multiple values map to the same index, place them together (using massage digest).

(4 Marks - [U/P, 2])

(iv) A cryptographic hash function comprising a thread of digits generated by using a one - way hashing formula. Message digets are set up to prevent the details of a piese of media or data and to identify changes and modifications to any part of a message. For example, While storing a file in any open cloud space, one would want to predict the integrity of the file. This problem can be solved using the message digest.

The following Python code constructs a hash table using the hash function
$h(x) = x mod$ 10h(x)=xmod10
to store the elements [4322, 1334, 1471, 9679, 1989, 6171, 6173, 4199] with collision handling using chaining (message digest method).

Identify missing code (1 ,2 , 3 and 4) to complete the program.

```
def hash_function(element, table_size=10):
    return element % table_size

elements = [4322, 1334, 1471, 9679, 1989, 6171, 6173, 4199]
hash_table = {}

for element in elements:
_________1_____________

    # Collision handling using chaining
    if hash_value in hash_table:
```

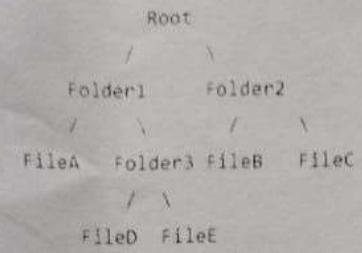

Figure : Binary Tree

Find the In order, Pre order and Post Order traversal for the above binary Tree .

(6 Marks - [Ap/C, 2])

C2	(i)	Consider a binary heap data structure, which is used for efficiently implementing priority queues. The binary heap is a complete binary tree where each parent node satisfies the heap property: either it is smaller than its children (min-heap) or larger than its children (max-heap). When inserting an element into the heap, the element is initially placed at the end of the tree, and then the heap property is restored by bubbling up the element.Select the time complexity of inserting an element into a binary heap a) O(1) b) O(log n) c) O(n) d) O(n log n) (1 Mark - [U/C, 2])
	(ii)	A file system uses a B-Tree to organize file directories and metadata for efficient access and updates. Unlike a binary search tree, the B-Tree is optimized for systems that read and write large blocks of data.Identify the main advantage of using a B-Tree over a binary search tree a) Faster insertion and deletion b) Better memory utilization c) Balanced tree structure, reducing height d) Increased storage capacity (1 Mark - [U/C, 2])
	(iii)	The flight reservation system needs to manage and retrieve the information about flights, such as the flight number, departure and arrival time, and ticket price. The flight number is stored using the tree data structure for efficient retrieval. In the tree, each node represents the flight number such as 50, 10, 60, 20,15, 8. Figure shows the unbalanced AVL tree. Find the balance factor and perform the necessary rotations to convert it into a balanced AVL tree

Figure : Binary Tree

Apply the concept of tree traversal techniques and mention the output for
a) Preorder traversal that shows a copy of the file system
b) Inorder traversal to print files and directories
c) Postorder traversal that makes deletion easier

(3 Marks - [Ap/C, 2])

(iv) In a hospital management system, the data of new patient records for each month is stored in a Binary Search Tree (BST). The tree structure allows efficient insertion and retrieval of patient records based on their admission date. Each patient record contains the patient's ID and the date of admission. The hospital tracks the number of new records each month and uses the BST to efficiently maintain this data. The following is the data representing the number of new patient records added for each month:

Monthly Patient Record Additions

Month	Patient Records Added
January	120
February	110
March	135
April	145
May	130
June	120
July	140
August	125
September	150
October	160
November	155
December	145

a) Find the average number of new patient records added per month, considering that the records are inserted into the BST in the order given.
b) Identify the month in which the highest increase in patient records was observed compared to the previous month, assuming that the records are inserted sequentially in the BST.
c) Find the month that had the lowest number of records added and explain its position in the BST structure.
d) Calculate the height of the BST if the records were inserted in the given order, and explain how the tree structure affects the performance of insertion and search operations.
e) Analyze the balance of the BST after all records have been inserted, and determine if rebalancing is needed to optimize the tree.

(5 Marks - [An/C, 2])