Paper ID : 260978

Subject Code:BOE406

Roll No:													

BTECH
(SEM. IV) THEORY EXAMINATION 2025-26
BASICS DATA STRUCTURE & ALGORITHMS

TIME: 3 HRS **M.MARKS: 70**

Note: Attempt all Sections. In case of any missing data; choose suitably.

SECTION A

1. Attempt *all* questions in brief. **02 x 7 = 14**

Qno.	Question	CO	Level
a.	List the characteristics of a good algorithm and briefly explain its role in real-life applications. एक अच्छे एल्गोरिथ्म की विशेषताओं की सूची बनाइए तथा वास्तविक जीवन (real-life) अनुप्रयोगों में इसकी भूमिका को संक्षेप में समझाइए।	5	K3
b.	What is the difference between array and linked stack? Array और Linked Stack के बीच क्या अंतर है?	1	K2
c.	Why is postfix evaluation easier than infix? Postfix अभिव्यक्ति का मूल्यांकन infix की तुलना में अधिक आसान क्यों होता है?	1	K2
d.	Differentiate between recursion and iteration. Recursion और Iteration के बीच अंतर स्पष्ट कीजिए।	4	K2
e.	Difference between adjacency matrix and list. Adjacency Matrix और Adjacency List के बीच अंतर बताइए।	3	K2
f.	What is minimum spanning tree? Minimum Spanning Tree क्या होता है?	3	K2
g.	How many labeled binary trees can be formed with N=3 nodes? N=3 नोड्स के साथ कितने लेबल्ड बाइनरी ट्री बनाए जा सकते हैं?	3	K3

SECTION B

2. Attempt any *three* of the following: **07 x 3 = 21**

a.	Differentiate among Big-O notation, Big-Theta notation, and Big-Omega notation. Also, define each notation with suitable examples. Big-O notation, Big-Theta notation और Big-Omega notation के बीच अंतर स्पष्ट कीजिए। साथ ही प्रत्येक notation को उपयुक्त उदाहरणों सहित परिभाषित कीजिए।	5	K3
b.	Convert the following infix expression into both prefix and postfix forms. Show all intermediate steps clearly, including the use of stack operations and operator precedence rules at each stage: A+(B*C-(D/E^F)*G)*H निम्नलिखित infix अभिव्यक्ति को prefix और postfix दोनों रूपों में परिवर्तित कीजिए। सभी मध्यवर्ती चरणों को स्पष्ट रूप से दिखाइए, जिसमें stack operations का उपयोग तथा प्रत्येक चरण पर operator precedence नियम शामिल हों। A+(B*C-(D/E^F)*G)*H	4	K3
c.	State the conditions for underflow and overflow in a circular queue. Then, for a circular queue of size 5, perform the following operations step by step, showing the position of front and rear after each operation and the final state of the queue: Enqueue(1), Enqueue(2), Enqueue(3), Dequeue(), Enqueue(4), Enqueue(5), Enqueue(6). सर्कुलर क्यू में underflow और overflow की स्थितियों (conditions) को बताइए। इसके बाद, आकार 5 (size 5) वाले एक सर्कुलर क्यू के लिए निम्नलिखित क्रियाएँ (operations) चरण-दर-चरण कीजिए, प्रत्येक ऑपरेशन के बाद front और rear की स्थिति तथा क्यू की अंतिम स्थिति स्पष्ट रूप से दिखाइए:	1	K3

QP26E_290 | 13-May-2026 2:35:15 PM | 117.55.242.134

Paper ID : 260978

Roll No:													

BTECH
(SEM. IV) THEORY EXAMINATION 2025-26
BASICS DATA STRUCTURE & ALGORITHMS

TIME: 3 HRS **M.MARKS: 70**

	Enqueue(1), Enqueue(2), Enqueue(3), Dequeue(), Enqueue(4), Enqueue(5), Enqueue(6)		
d.	Compare Insertion, Selection, and Quick Sort based on time complexity, stability, and adaptability. Insertion Sort, Selection Sort और Quick Sort की तुलना समय जटिलता (time complexity), स्थिरता (stability) तथा अनुकूलता (adaptability) के आधार पर कीजिए।	2	K2
e.	Insert the given elements into a Binary Search Tree (BST) using two different orders: (i) Sorted order: 10, 20, 30, 40, 50 (ii) Random order: 30, 10, 50, 20, 40 Construct the BST for both cases step by step. Then, determine the height of each tree and compare their structures. दिए गए तत्वों को Binary Search Tree (BST) में दो विभिन्न क्रमों में insert कीजिए: (i) क्रमबद्ध क्रम (sorted order): 10, 20, 30, 40, 50 (ii) (ii) यादृच्छिक क्रम (random order): 30, 10, 50, 20, 40। दोनों मामलों में BST का चरण-दर-चरण निर्माण कीजिए। इसके बाद प्रत्येक वृक्ष की ऊँचाई (height) ज्ञात कीजिए तथा उनकी संरचना (structure) की तुलना कीजिए।	3	K3

SECTION C

3. Attempt any *one* part of the following: **07 x 1 = 07**

a.	Derive the general formula to calculate the address of an element A[i][j] in column-major order, and illustrate the process with a suitable example. Column-major क्रम में A[i][j] तत्व का पता (address) निकालने के लिए सामान्य सूत्र (general formula) व्युत्पन्न (derive) कीजिए, तथा उपयुक्त उदाहरण की सहायता से प्रक्रिया को समझाइए।	1	K2
b.	Write algorithms for Depth First Search (DFS), explaining steps, data structures used, and illustrating their working with suitable examples. Depth First Search (DFS) के लिए एल्गोरिथ्म लिखिए, जिसमें चरणों (steps), उपयोग किए जाने वाले डेटा स्ट्रक्चर्स (data structures) की व्याख्या कीजिए तथा उपयुक्त उदाहरणों की सहायता से इसके कार्य (working) को समझाइए।	3	K2

4. Attempt any *one* part of the following: **07 x 1 = 07**

a.	Perform the following operations on a stack of size 5: Push(10), Push(20), Push(30), Pop(), Push(40), Push(50), Push(60). Show stack after each step and identify overflow/underflow. आकार 5 (size 5) वाले एक स्टैक पर निम्नलिखित क्रियाएँ (operations) कीजिए: Push(10), Push(20), Push(30), Pop(), Push(40), Push(50), Push(60)। प्रत्येक चरण के बाद स्टैक की स्थिति (state) दिखाइए तथा overflow/underflow की पहचान कीजिए।	4	K3
b.	Insert keys 19, 29, 39, 49, 59 into hash table (size = 10) using h(k) = k mod 10. Resolve collisions using linear probing. आकार 10 (size = 10) वाली हैश टेबल में कुंजियाँ (keys) 19, 29, 39, 49, 59 को h(k) = k mod 10 हैश फंक्शन का उपयोग करके insert कीजिए। टकराव (collisions) को linear probing द्वारा हल कीजिए।	2	K3

QP26E_290 | 13-May-2026 2:35:15 PM | 117.55.242.134

QP26E_290 | 13-May-2026 2:35:15 PM | 117.55.242.134

Paper ID : 260978

Subject Code:BOE406

Roll No:													

BTECH
(SEM. IV) THEORY EXAMINATION 2025-26
BASICS DATA STRUCTURE & ALGORITHMS

TIME: 3 HRS **M.MARKS: 70**

---

5. Attempt any *one* part of the following: 07 x 1 = 07

a.	Given a sorted array A = {5, 9, 14, 21, 28, 35, 42, 49, 56}, perform binary search to find the element 35. Show each step of the search process, including the calculation of mid index and comparison at every iteration. Also, analyze the time complexity of the algorithm. दिए गए sorted array A = {5, 9, 14, 21, 28, 35, 42, 49, 56} में तत्व 35 को खोजने के लिए binary search का उपयोग कीजिए। प्रत्येक चरण में search प्रक्रिया को दिखाइए, जिसमें प्रत्येक iteration में mid index की गणना और तुलना शामिल हो। साथ ही एल्गोरिथ्म की समय जटिलता (time complexity) का विश्लेषण कीजिए।	2	K3
b.	Explain Prim's algorithm and Kruskal's algorithm for finding a minimum spanning tree with suitable real-life examples. Compare their approaches, steps, and efficiency. Minimum Spanning Tree खोजने के लिए Prim's algorithm और Kruskal's algorithm को उपयुक्त वास्तविक जीवन (real-life) उदाहरणों सहित समझाइए। इनके दृष्टिकोण (approaches), चरणों (steps) और दक्षता (efficiency) की तुलना कीजिए।	5	K2

6. Attempt any *one* part of the following: 07 x 1 = 07

a.	Convert recursive factorial function into iterative version and trace for n = 5. Recursive factorial function को iterative रूप (iterative version) में परिवर्तित कीजिए तथा n = 5 के लिए उसका trace कीजिए।	4	K3
b.	Construct a binary tree from: Preorder: A B D H I E C F G Inorder: H D I B E A F C G निम्नलिखित से एक बाइनरी ट्री (binary tree) का निर्माण कीजिए: Preorder: A B D H I E C F G Inorder: H D I B E A F C G	3	K3

7. Attempt any *one* part of the following: 07 x 1 = 07

a.	Explain how to represent a polynomial using a linked list data structure. Illustrate your answer by representing the polynomial $3x^2 + 5x + 2$ using a linked list, and describe the structure of each node along with a suitable diagram. Linked list data structure का उपयोग करके एक बहुपद को कैसे प्रदर्शित किया जाता है, इसे समझाइए। अपने उत्तर को $3x^2 + 5x + 2$ बहुपद को linked list के रूप में प्रदर्शित करके स्पष्ट कीजिए, तथा प्रत्येक node की संरचना का वर्णन करते हुए एक उपयुक्त आरेख (diagram) भी दीजिए।	1	K2
b.	Apply the Quick Sort algorithm to the array [10, 80, 30, 90, 40, 50, 70] by selecting the first element as the pivot, and clearly illustrate the partitioning process at each step. पहले तत्व (first element) को pivot के रूप में चुनकर [10, 80, 30, 90, 40, 50, 70] array पर Quick Sort एल्गोरिथ्म लागू कीजिए तथा प्रत्येक चरण में partitioning प्रक्रिया को स्पष्ट रूप से दर्शाइए।	2	K3

QP26E_290 | 13 May 2026 2:35:15 PM | 117.55.242.134

QP26E_290 | 13-May-2026 2:35:15 PM | 117.55.242.134