Roll No:-_____

(Sem-3_OBE) Diploma Exam 2025 (Odd) Supp/Comp

[Time: 3 Hours] **Artificial Intelligence (AI) / Artificial Intelligence (AI) and Machine Learning / C.S.E / Computer Science / Information Science**
(Theory)
Data Structures and Algorithms
(T2418301) **[Full. Marks: 70]**

- All questions are compulsory. (सभी प्रश्न अनिवार्य है।)

- Marks are mentioned on the right side of each question. (अंक सभी प्रश्न के दाई ओर अंकित किये है।)

Group (A) (ग्रुप -ए)

Q.1 Answer all questions as directed. **(2x10=20)**
(निर्देशानुसार सभी प्रश्नों के उत्तर दें)

	Marks	CO	BL
a) The time complexity of binary search is: (a) O(n) (b) O(log n) (c) O(n^2) (d) O(1) बाइनरी सर्च की समय जटिलता क्या है? (a) O(n) (b) O(log n) (c) O(n^2) (d) O(1)	2	1	1
b) A _____,follows the LIFO principle. (Stack/Queue/Linked List) _____, LIFO सिद्धांत का पालन करता है। (स्टैक/क्यू/लिंक्ड लिस्ट)	2	2	1
c) In a circular queue, the next position after the last element is _____. (first/middle/last) सर्कुलर क्यू में, अंतिम तत्व के बाद अगली स्थिति _____ है। (पहली/मध्य/अंतिम)	2	2	1
d) **Match the Following:** **Data Structure** — **Description** A. Stack — 1. FIFO principle B. Queue — 2. LIFO principle C. Array — 3. Dynamic memory allocation D. Linked List — 4. Contiguous memory **मिलान करें –** **डेटा संरचना** — **विवरण** A. स्टैक — 1. FIFO सिद्धांत B. क्यू — 2. LIFO सिद्धांत C. एरे — 3. डायनामिक मेमोरी एलोकेशन D. लिंक्ड लिस्ट — 4. कंटिग्यूस मेमोरी	2	2	2
(e) An application of a stack is _____. (Function call management/ BFS traversal/ Sorting array/ Minimum spanning tree) स्टैक का एक एप्लीकेशन _____ है। (फ़ंक्शन कॉल प्रबंधन/ BFS ट्रैवर्सल/ एरे सॉर्टिंग/ न्यूनतम स्पैनिंग ट्री)	2	2	1
(f) In a Max Heap, the root node contains: (a) Minimum element (b) Maximum element (c) Random element (d) Median element मैक्स हीप में, रूट नोड में कौन सा तत्व होता है: (a) न्यूनतम तत्व (b) अधिकतम तत्व (c) यादृच्छिक तत्व (d) माध्यक तत्व	2	3	2
(g) In a priority queue, elements are dequeued randomly.(**True/False)**	2	3	1

प्रायोरिटी क्यू में तत्व यादृच्छिक रूप से निकाले जाते हैं।**(सही/गलत)**			
(h) Dynamic programming solves overlapping subproblems efficiently.(**True/False)** डायनामिक प्रोग्रामिंग ओवरलैपिंग सबप्रॉब्लम्स को कुशलतापूर्वक हल करता है।**(सही/गलत)**	2	5	1
(i) **Match the following:** **Algorithm** — **Feature** A. Greedy — 1. Optimal substructure B. Divide & Conquer — 2. Recursion-based solution C. Dynamic Programming — 3. Overlapping subproblems D. BFS — 4. Graph traversal **मिलान करें** – **एल्गोरिथम** — **गुण** A. ग्रेडी — 1. ऑप्टिमल सबस्ट्रक्चर B. डिवाइड एंड कॉन्कर — 2. रिकर्शन-आधारित समाधान C. डायनामिक प्रोग्रामिंग — 3. ओवरलैपिंग सबप्रॉब्लम्स D. BFS — 4. ग्राफ ट्रैवर्सल	2	5	2
(j) Quick Sort is based on _____ technique. (Divide and Conquer/Greedy/Dynamic Programming) क्विक सॉर्ट _____ तकनीक पर आधारित है। (डिवाइड एंड कॉन्कर/ग्रेडी/डायनामिक प्रोग्रामिंग)	2	4	1
Group (B) (ग्रुप -बी)			
Answer all five questions. (सभी पाँच प्रश्नों के उत्तर दें।) **4x5=20**			
Q.2 Compare best-case, worst-case, and average-case time complexity with examples. सर्वश्रेष्ठ, सबसे खराब और औसत समय जटिलताओं की तुलना उदाहरण सहित करें।	4	1	2
OR (अथवा)			
Distinguish between time complexity and space complexity of an algorithm. एल्गोरिथ्म की समय जटिलता और स्थान जटिलता में अंतर स्पष्ट कीजिए।	4	1	2
Q.3 Explain the advantages of linked list over arrays in dynamic memory allocation. डायनामिक मेमोरी एलोकेशन में एरे की तुलना में लिंक्ड लिस्ट के फायदों के बारे में बताएं।	4	2	1
OR (अथवा)			
Explain a circular queue and its advantages over a linear queue. सर्कुलर क्यू को समझाइए और लीनियर क्यू पर इसके लाभ बताएँ।	4	2	1
Q.4 Explain the time and space complexity of basic stack and queue operations. बेसिक स्टैक और क्यू ऑपरेशन्स की टाइम और स्पेस कॉम्प्लेक्सिटी के बारे में बताएं।	4	2	1
OR (अथवा)			
Explain the basic terminologies of trees with examples. ट्री की बुनियादी शब्दावली उदाहरण सहित समझाइए।	4	3	1
Q.5 Explain BFS and DFS traversal techniques with examples. उदाहरणों के साथ BFS और DFS ट्रैवर्सल तकनीकों को समझाएँ।	4	3	1
OR (अथवा)			
Explain different tree traversal techniques. अलग-अलग ट्री ट्रैवर्सल तकनीकों के बारे में बताएं।	4	3	1
Q.6 Explain linear search algorithms. लीनियर सर्च एल्गोरिथ्म को समझाइए।	4	4	1
OR (अथवा)			
Explain the greedy choice property with an example. उदाहरण सहित ग्रीडी चॉइस प्रॉपर्टी को समझाइए।	4	5	1

Group (C) (ग्रुप - सी)

Answer all five questions. (सभी पाँच प्रश्नों के उत्तर दें |) **6x5=30**

Q.7	Write an algorithm to find the largest element in an array and discuss its time complexity. एक एल्गोरिथ्म लिखिए जो एक एरे में सबसे बड़ा तत्व खोजे और इसके समय जटिलता पर चर्चा कीजिए।	6	2	3
	OR (अथवा)			
	Write the algorithm for reversing a stack using recursion. पुनरावृत्ति का उपयोग करके स्टैक को उलटने के लिए एल्गोरिथ्म लिखिए।	6	2	3
Q.8	Explain Big-O, Omega, and Theta notations with suitable examples. Big-O, Omega और Theta संकेतन को उपयुक्त उदाहरणों के साथ समझाइए।	6	1	1
	OR (अथवा)			
	Explain the process of insertion and deletion in a linked list. लिंक्ड लिस्ट में तत्व डालने और हटाने की प्रक्रिया समझाइए।	6	2	1
Q.9	Explain Dijkstra's algorithm for shortest path in weighted graphs. वेटेड ग्राफ में सबसे छोटी दूरी के लिए Dijkstra एल्गोरिथ्म समझाइए।	6	3	1
	OR (अथवा)			
	Explain the concept of heap and its types. हीप की अवधारणा और इसके समझाइए।	6	3	1
Q.10	Compare selection sort and insertion sort in terms of efficiency. दक्षता के संदर्भ में चयन सॉर्ट और इन्सर्शन सॉर्ट की तुलना करें।	6	4	2
	OR (अथवा)			
	Perform Bubble Sort on the array [8, 3, 5, 2, 7] and show all passes. एरे [8, 3, 5, 2, 7] पर बबल सॉर्ट लागू कीजिए और सभी पास दिखाइए।	6	4	3
Q.11	Explain top-down and bottom-up approaches in dynamic programming. डायनेमिक प्रोग्रामिंग में टॉप-डाउन और बॉटम-अप दृष्टिकोण समझाइए।	6	5	1
	'OR (अथवा)			
	Apply Insertion Sort on the array [7, 2, 5, 3] and illustrate each insertion step. एरे [7, 2, 5, 3] पर इन्सर्शन सॉर्ट लागू कीजिए और प्रत्येक इन्सर्शन स्टेप दिखाइए।	6	4	3