

JK

Roll No. _____

Question Booklet Number

3560230

O.M.R. Serial No. :

--	--	--	--	--	--	--

BCA II Semester Examination, 2025-26

Basics of Data Structure and Algorithms

Paper Code

B	C	A	2	0	0	2	T
---	---	---	---	---	---	---	---

Question Booklet Series

B

Time : 1 : 30 Hours]

[Maximum Marks : 75

Instructions to the Examinee :

1. Do not open the booklet unless you are asked to do so.
2. The booklet contains 100 questions. Examinee is required to answer 75 questions in the OMR Answer-Sheet provided and not in the question booklet. All questions carry equal marks.
3. Examine the Booklet and the OMR Answer-Sheet very carefully before you proceed. Faulty question booklet due to missing or duplicate pages/questions or having any other discrepancy should be got immediately replaced.
4. Four alternative answers are mentioned for each question as – A, B, C & D in the booklet. The candidate has to choose the correct answer and mark the same in the OMR Answer-Sheet as per the direction :

(Remaining instructions on the last page)

परीक्षार्थियों के लिए निर्देश :

1. प्रश्न-पुस्तिका को तब तक न खोलें जब तक आपसे कहा न जाए।
2. प्रश्न-पुस्तिका में 100 प्रश्न हैं। परीक्षार्थी को 75 प्रश्नों के केवल दी गई OMR आन्सर-शीट पर ही हल करना है, प्रश्न-पुस्तिका पर नहीं। सभी प्रश्नों के अंक समान हैं।
3. प्रश्नों के उत्तर अंकित करने से पूर्व प्रश्न-पुस्तिका तथा OMR आन्सर-शीट को सावधानीपूर्वक देख लें। दोषपूर्ण प्रश्न-पुस्तिका जिसमें कुछ भाग छपने से छूट गये हों या प्रश्न एक से अधिक बार छप गए हों या उसमें किसी अन्य प्रकार की कमी हो, तो उसे तुरन्त बदल लें।
4. प्रश्न-पुस्तिका में प्रत्येक प्रश्न के चार सम्भावित उत्तर- A, B, C तथा D हैं। परीक्षार्थी को उन चारों विकल्पों में से सही उत्तर छाँटना है। उत्तर को OMR उत्तर-पत्रक में सम्बन्धित प्रश्न संख्या में निम्न प्रकार भरना है :

(शेष निर्देश अन्तिम पृष्ठ पर)

1. Which complexity is most efficient?
 - (A) $O(n^2)$
 - (B) $O(n \log n)$
 - (C) $O(\log n)$
 - (D) $O(2^n)$
2. Stack top refers to:
 - (A) First element
 - (B) Last inserted
 - (C) Middle
 - (D) Random
3. Queue rear indicates:
 - (A) Deletion
 - (B) Insertion
 - (C) Middle
 - (D) Random
4. Binary tree max nodes at level L:
 - (A) L^2
 - (B) 2^L
 - (C) L
 - (D) 2L
5. Internal node has:
 - (A) No child
 - (B) At least one child
 - (C) One child
 - (D) Two children
6. Subtree is:
 - (A) Part of tree
 - (B) Entire tree
 - (C) Node
 - (D) Edge
7. Degree of node means:
 - (A) Number of children
 - (B) Number of parents
 - (C) Number of edges
 - (D) Height
8. DFS traversal order depends on:
 - (A) Stack
 - (B) Queue
 - (C) Array
 - (D) Tree

9. BFS traversal order depends on:
- (A) Stack
 - (B) Queue
 - (C) Array
 - (D) Tree
10. Adjacency matrix diagonal represents:
- (A) Self loop
 - (B) Edge
 - (C) Node
 - (D) Path
11. Algorithm efficiency depends on:
- (A) Input size
 - (B) Output size
 - (C) Compiler
 - (D) Language
12. Which is constant time?
- (A) $O(1)$
 - (B) $O(n)$
 - (C) $O(n^2)$
 - (D) $O(\log n)$
13. Worst case means:
- (A) Minimum time
 - (B) Maximum time
 - (C) Average time
 - (D) Constant time
14. Queue underflow occurs when:
- (A) Full
 - (B) Empty
 - (C) Half
 - (D) Sorted
15. Stack underflow occurs when:
- (A) Full
 - (B) Empty
 - (C) Half
 - (D) Sorted
16. Array is stored in:
- (A) Random memory
 - (B) Contiguous memory
 - (C) Stack
 - (D) Queue

17. Linked list memory is:
- (A) Contiguous
 - (B) Non-contiguous
 - (C) Fixed
 - (D) Static
18. Circular queue rear connects to:
- (A) Front
 - (B) Middle
 - (C) Null
 - (D) Itself
19. Prefix evaluation uses:
- (A) Queue
 - (B) Stack
 - (C) Array
 - (D) Tree
20. Postfix evaluation uses:
- (A) Queue
 - (B) Stack
 - (C) Tree
 - (D) Graph
21. Graph traversal ensures:
- (A) Visiting all nodes
 - (B) Sorting nodes
 - (C) Deleting nodes
 - (D) Searching edges
22. Binary search requires:
- (A) Sorted array
 - (B) Unsorted array
 - (C) Linked list
 - (D) Stack
23. Height-balanced tree improves:
- (A) Search time
 - (B) Insert time
 - (C) Delete time
 - (D) Space
24. Full binary tree property:
- (A) 0 or 2 children
 - (B) Only 1 child
 - (C) 3 children
 - (D) Infinite
25. Complete binary tree is:
- (A) Fully filled
 - (B) Filled except last level
 - (C) Empty
 - (D) Random

26. Tree representation using array is efficient for:
- (A) Complete tree
 - (B) Sparse tree
 - (C) Random tree
 - (D) Linked tree
27. Linked representation uses:
- (A) Pointers
 - (B) Index
 - (C) Values
 - (D) Arrays
28. DFS explores:
- (A) Level-wise
 - (B) Depth-wise
 - (C) Random
 - (D) Reverse
29. BFS explores:
- (A) Depth-wise
 - (B) Level-wise
 - (C) Random
 - (D) Reverse
30. Graph cycle means:
- (A) Path returning to start
 - (B) Single node
 - (C) Edge only
 - (D) Tree
31. Time complexity of BFS is:
- (A) $O(V+E)$
 - (B) $O(n^2)$
 - (C) $O(\log n)$
 - (D) $O(1)$
32. Time complexity of DFS is:
- (A) $O(V+E)$
 - (B) $O(n^2)$
 - (C) $O(\log n)$
 - (D) $O(1)$
33. Stack implementation can be:
- (A) Array
 - (B) Linked list
 - (C) Both
 - (D) None

34. Queue implementation can be:

- (A) Array
- (B) Linked list
- (C) Both
- (D) None

35. Dynamic structure is:

- (A) Array
- (B) Linked list
- (C) Matrix
- (D) Table

36. Static structure is:

- (A) Linked list
- (B) Array
- (C) Graph
- (D) Tree

37. Which traversal uses recursion naturally?

- (A) BFS
- (B) DFS
- (C) Queue
- (D) Stack

38. Tree edge count =

- (A) Nodes
- (B) Nodes - 1
- (C) Nodes + 1
- (D) Nodes²

39. Graph with no cycles is:

- (A) Tree
- (B) Directed
- (C) Undirected
- (D) Weighted

40. Minimum edges in tree with n nodes:

- (A) n
- (B) n-1
- (C) n+1
- (D) n²

41. Maximum edges in complete graph:

- (A) n
- (B) $n(n-1)/2$
- (C) n²
- (D) n+1

42. Path length means:

- (A) Number of edges
- (B) Nodes
- (C) Leaves
- (D) Height

43. Stack pointer indicates:

- (A) Top
- (B) Bottom
- (C) Middle
- (D) Random

44. Queue pointer front indicates:

- (A) Deletion
- (B) Insertion
- (C) Middle
- (D) Random

45. Balanced BST improves:

- (A) $O(n)$ to $O(\log n)$
- (B) $O(\log n)$ to $O(n)$
- (C) $O(n^2)$ to $O(n)$
- (D) No change

46. Searching in skewed BST is:

- (A) $O(\log n)$
- (B) $O(n)$
- (C) $O(1)$
- (D) $O(n^2)$

47. Which is best for undo operations?

- (A) Queue
- (B) Stack
- (C) Tree
- (D) Graph

48. Which structure is best for scheduling?

- (A) Stack
- (B) Queue
- (C) Tree
- (D) Graph

49. Which is non-linear?

- (A) Array
- (B) Stack
- (C) Queue
- (D) Tree

50. Which is linear?

- (A) Tree
- (B) Graph
- (C) Stack
- (D) Heap

51. Which of the following best describes an Abstract Data Type (ADT)?

- (A) Memory layout of data
- (B) Set of operations without implementation
- (C) Data stored in array
- (D) Algorithm complexity

52. Which complexity grows fastest for large input size?

- (A) $O(n)$
- (B) $O(\log n)$
- (C) $O(n^2)$
- (D) $O(1)$

53. In Big-O notation, constants are ignored because:

- (A) They increase complexity
- (B) They are always zero
- (C) They do not affect asymptotic growth
- (D) They are variables

54. Which is NOT a property of an algorithm?

- (A) Finiteness
- (B) Ambiguity
- (C) Definiteness
- (D) Input

55. Which structure is inherently non-linear?

- (A) Array
- (B) Stack
- (C) Queue
- (D) Graph

56. In row-major order, the next element after $A[i][j]$ is:

- (A) $A[i+1][j]$
- (B) $A[i][j+1]$
- (C) $A[j][i]$
- (D) $A[i-1][j]$

57. Sparse matrix representation reduces:

- (A) Time
- (B) Space
- (C) Complexity
- (D) Operations

58. Accessing an element in an array takes:
- (A) $O(n)$
 - (B) $O(\log n)$
 - (C) $O(1)$
 - (D) $O(n^2)$
59. Worst case insertion in array requires:
- (A) No shift
 - (B) One shift
 - (C) Multiple shifts
 - (D) No operation
60. Linked list nodes are connected using:
- (A) Index
 - (B) Pointer
 - (C) Value
 - (D) Key
61. Which list allows backward traversal?
- (A) Singly
 - (B) Doubly
 - (C) Circular
 - (D) Linear
62. In circular linked list, last node points to:
- (A) NULL
 - (B) First node
 - (C) Middle node
 - (D) Itself
63. Which structure is best for function calls?
- (A) Queue
 - (B) Stack
 - (C) Tree
 - (D) Graph
64. Stack overflow occurs when:
- (A) Empty
 - (B) Full
 - (C) Half full
 - (D) Sorted
65. Infix to postfix conversion uses:
- (A) Queue
 - (B) Stack
 - (C) Tree
 - (D) Graph

66. Which expression is postfix?
- (A) $A+B$
 - (B) $+AB$
 - (C) $AB+$
 - (D) $BA+$
67. Queue follows:
- (A) LIFO
 - (B) FIFO
 - (C) FILO
 - (D) Random
68. Enqueue operation inserts at:
- (A) Front
 - (B) Rear
 - (C) Middle
 - (D) Random
69. Dequeue removes from:
- (A) Rear
 - (B) Front
 - (C) Middle
 - (D) Random
70. Circular queue avoids:
- (A) Underflow
 - (B) Overflow
 - (C) Wastage of space
 - (D) Sorting
71. Priority queue removes:
- (A) First inserted
 - (B) Last inserted
 - (C) Highest priority
 - (D) Random
72. Dequeue allows insertion at:
- (A) One end
 - (B) Two ends
 - (C) Middle
 - (D) Random
73. Tree root has:
- (A) Parent
 - (B) No parent
 - (C) Two parents
 - (D) Infinite parents
74. Leaf node has:
- (A) One child
 - (B) No child
 - (C) Two children
 - (D) Infinite
75. Binary tree max children per node:
- (A) 1
 - (B) 2
 - (C) 3
 - (D) Infinite

76. Inorder traversal sequence is:

- (A) Root-Left-Right
- (B) Left-Root-Right
- (C) Left-Right-Root
- (D) Right-Left-Root

77. Preorder traversal sequence is:

- (A) Root-Left-Right
- (B) Left-Root-Right
- (C) Left-Right-Root
- (D) Right-Root-Left

78. Postorder traversal sequence is:

- (A) Root-Left-Right
- (B) Left-Root-Right
- (C) Left-Right-Root
- (D) Root-Right-Left

79. BST property ensures:

- (A) Sorted traversal
- (B) Random traversal
- (C) Reverse traversal
- (D) No traversal

80. BST search time depends on:

- (A) Nodes
- (B) Height
- (C) Leaves
- (D) Edges

81. Which traversal is used to delete a tree?

- (A) Preorder
- (B) Inorder
- (C) Postorder
- (D) BFS

82. DFS uses:

- (A) Queue
- (B) Stack
- (C) Tree
- (D) Array

83. BFS uses:

- (A) Stack
- (B) Queue
- (C) Tree
- (D) Array

84. Adjacency matrix requires:
- (A) $O(n)$
 - (B) $O(n^2)$
 - (C) $O(\log n)$
 - (D) $O(1)$
85. Graph vertex represents:
- (A) Edge
 - (B) Node
 - (C) Path
 - (D) Cycle
86. Edge connects:
- (A) Nodes
 - (B) Trees
 - (C) Arrays
 - (D) Lists
87. Directed graph has:
- (A) No direction
 - (B) Direction
 - (C) One node
 - (D) No edge
88. Undirected graph has:
- (A) Direction
 - (B) No direction
 - (C) One node
 - (D) No edge
89. Path in graph is:
- (A) Sequence of vertices
 - (B) Single vertex
 - (C) Edge only
 - (D) Random
90. Sparse graph is better represented by:
- (A) Matrix
 - (B) List
 - (C) Stack
 - (D) Queue
91. Time complexity of linear search:
- (A) $O(1)$
 - (B) $O(\log n)$
 - (C) $O(n)$
 - (D) $O(n^2)$

92. Best case of binary search:
- (A) $O(1)$
 - (B) $O(n)$
 - (C) $O(\log n)$
 - (D) $O(n^2)$
93. Stack is used in:
- (A) BFS
 - (B) DFS
 - (C) Sorting
 - (D) Searching
94. Queue is used in:
- (A) DFS
 - (B) BFS
 - (C) Sorting
 - (D) Searching
95. Linked list insertion is:
- (A) Costly
 - (B) Efficient
 - (C) Impossible
 - (D) Constant always
96. Array deletion requires:
- (A) No shift
 - (B) Shifting elements
 - (C) Pointer change
 - (D) Sorting
97. Node contains:
- (A) Data only
 - (B) Pointer only
 - (C) Data + pointer
 - (D) Nothing
98. Doubly linked list uses:
- (A) One pointer
 - (B) Two pointers
 - (C) No pointer
 - (D) Three pointers
99. Circular list has:
- (A) No end
 - (B) One end
 - (C) Two ends
 - (D) Infinite ends
100. Tree height is:
- (A) Nodes count
 - (B) Levels count
 - (C) Leaves
 - (D) Edges