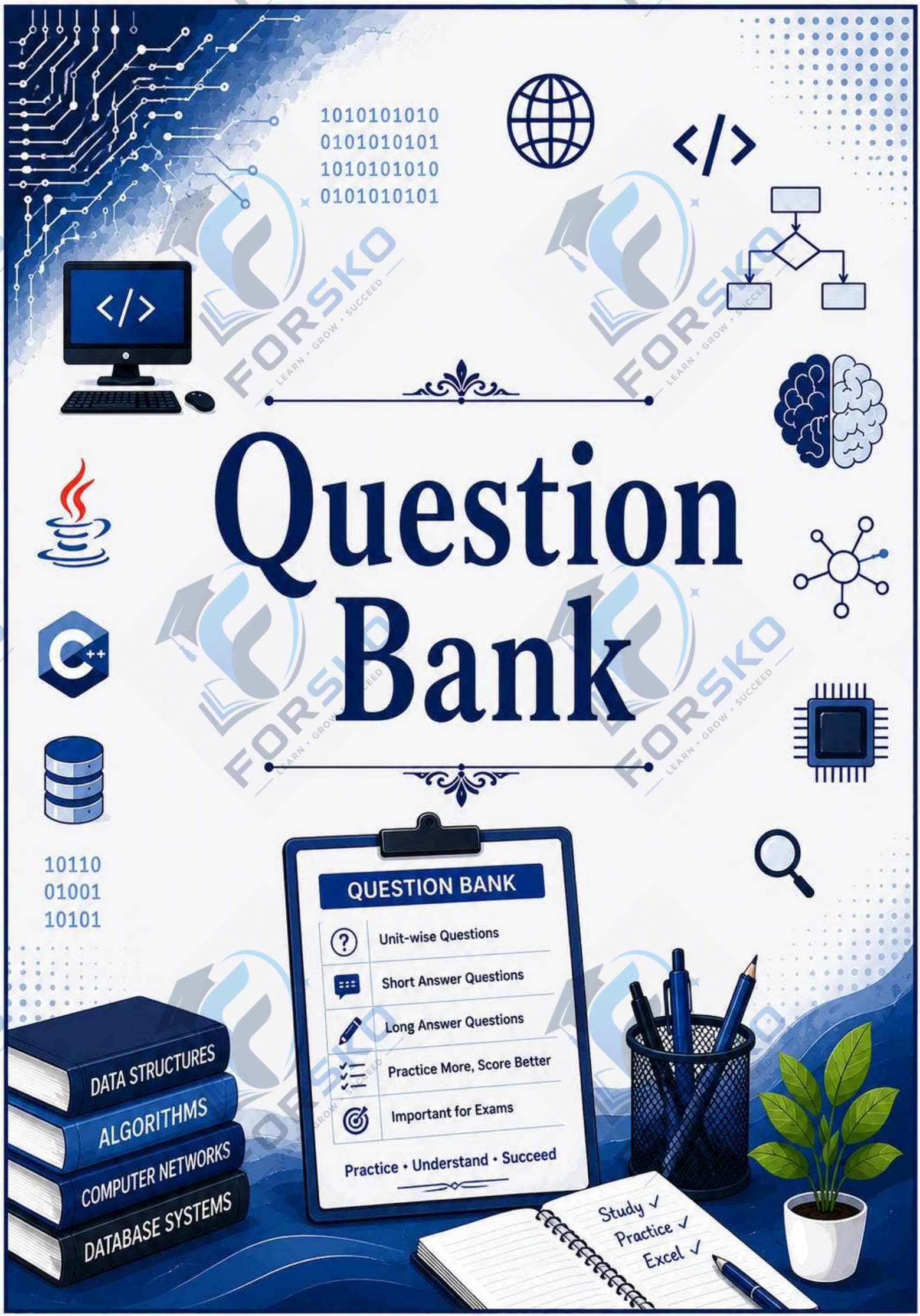
1010101010
0101010101
1010101010
0101010101
Question Bank
10110
01001
10101
QUESTION BANK
Unit-wise Questions
Short Answer Questions
Long Answer Questions
Practice More, Score Better
Important for Exams
Practice • Understand • Succeed
DATA STRUCTURES
ALGORITHMS
COMPUTER NETWORKS
DATABASE SYSTEMS
Study ✓
Practice ✓
Excel ✓

Data Structure

Unit I - Data Structure and Arrays

- **Answers in one or two lines (1 mark questions)**

1. Define Data Structure.
2. What is a primitive data structure?
3. What is a non-primitive data structure?
4. What is a linear data structure?
5. What is a non-linear data structure?
6. Name any two primitive data structures.
7. Name any two non-primitive data structures.
8. Define an array.
9. What is array traversal?
10. What is array insertion?
11. What is array deletion?
12. What is the memory representation of an array?
13. What is an index in an array?
14. What is the first index of an array in C?
15. What is a one-dimensional array?
16. What is a two-dimensional array?
17. Which data structure stores homogeneous elements?
18. What is the size of an array?
19. Name any three operations performed on arrays.
20. Which data structure stores data in contiguous memory locations?

- **Short Answer Questions (3 marks questions)**

1. Explain the concept of a data structure.
2. Explain the types of data structures with examples.
3. Differentiate between primitive and non-primitive data structures.
4. Differentiate between linear and non-linear data structures.
5. Explain various operations performed on data structures.
6. Define an array and explain its characteristics.
7. Explain the memory representation of a one-dimensional array.
8. Explain array traversal with an example.
9. Explain array insertion with an example.
10. Explain array deletion with an example.
11. Explain the advantages of arrays.
12. Explain the disadvantages of arrays.
13. Differentiate between one-dimensional and two-dimensional arrays.

- **Long Answer Questions (6 marks questions)**

1. Define Data Structure and explain its types with suitable examples.
2. Explain primitive and non-primitive data structures in detail with examples.
3. Differentiate between linear and non-linear data structures with a comparison table.
4. Explain arrays, their characteristics, and memory representation with suitable diagrams.
5. Explain array traversal, insertion, and deletion with suitable algorithms and examples.
6. Explain the memory representation of one-dimensional arrays with examples.
7. Explain the advantages, disadvantages, and applications of arrays.

Unit II -Stacks and Queues

- **Answers in one or two lines (1 mark questions)**

1. Define a stack.
2. What is a queue?
3. Which principle does a stack follow?
4. Which principle does a queue follow?
5. Expand LIFO.
6. Expand FIFO.
7. What is the PUSH operation?
8. What is the POP operation?
9. What is the ENQUEUE operation?
10. What is the DEQUEUE operation?
11. What is stack overflow?
12. What is stack underflow?
13. What is queue overflow?
14. What is queue underflow?
15. What is the TOP pointer in a stack?
16. What is the FRONT pointer in a queue?
17. What is the REAR pointer in a queue?
18. Name any two types of queues.
19. What is a circular queue?
20. What is a priority queue?

- **Short Answer Questions (3 marks questions)**

1. Explain the concept of a stack.
2. Explain the memory representation of a stack.
3. Explain the PUSH operation with an example.
4. Explain the POP operation with an example.
5. Explain stack traversal.
6. Explain stack overflow and stack underflow.
7. Explain the concept of a queue.
8. Explain the memory representation of a queue.
9. Explain the ENQUEUE operation with an example.
10. Explain the DEQUEUE operation with an example.
11. Explain queue traversal.
12. Differentiate between stack and queue.
13. Explain a circular queue.
14. Explain priority queue.

- **Long Answer Questions (6 marks questions)**

1. Define a stack and explain its memory representation with suitable diagrams.
2. Explain stack operations (Push, Pop, Traversing) with suitable algorithms.
3. Explain queue and its memory representation with suitable diagrams.
4. Explain queue operations (Enqueue, Dequeue, Traversing) with suitable algorithms.
5. Differentiate between stack and queue with a comparison table and examples.
6. Explain different types of queues with suitable diagrams.
7. Explain stack overflow and stack underflow with examples.
8. Explain queue overflow and queue underflow with examples.
9. Explain the applications of stacks and queues in computer science.
10. Write algorithms for Push, Pop, Enqueue, and Dequeue operations.

Unit III -Linked List and Trees

- **Answers in one or two lines (1 mark questions)**

1. Define a linked list.
2. What is a node in a linked list?
3. What is a pointer?
4. What is the memory representation of a linked list?
5. Name any two types of linked lists.
6. What is a singly linked list?
7. What is a doubly linked list?
8. What is a circular linked list?
9. What is linked list traversal?
10. What is linked list insertion?
11. What is linked list deletion?
12. Define a tree.
13. Define the root node.
14. Define a leaf node.
15. What is the level of a node?
16. What is the height of a tree?
17. What is a binary tree?
18. What is a complete binary tree?
19. What is a binary search tree (BST)?
20. Name the three tree traversal methods.

- **Short Answer Questions (3 marks questions)**

1. Explain the concept of a linked list.
2. Explain the memory representation of a linked list.
3. Explain different types of linked lists.
4. Explain linked list traversal.
5. Explain insertion in a linked list.
6. Explain deletion in a linked list.
7. Differentiate between an array and a linked list.
8. Explain the basic terminologies of a tree.
9. Explain the memory representation of a tree.
10. Explain different types of trees.
11. Explain binary trees with examples.
12. Explain binary search trees (BST).
13. Explain preorder tree traversal.
14. Explain inorder and postorder tree traversals.
15. Explain insertion and deletion in a binary search tree.

- **Long Answer Questions (6 marks questions)**

1. Explain linked lists with suitable diagrams and memory representation.
2. Explain different types of linked lists with suitable diagrams.
3. Explain linked list operations (Traversal, Insertion, Deletion) with algorithms.
4. Differentiate between arrays and linked lists with suitable examples.
5. Explain tree terminologies with neat diagrams.
6. Explain different types of trees with suitable examples.
7. Explain binary trees, complete binary trees, and binary search trees in detail.
8. Explain preorder, inorder, and postorder traversals with suitable examples.
9. Explain insertion and deletion operations in a binary search tree with algorithms.
10. Write a detailed note on the applications of linked lists and trees.

Unit IV - Searching and Sorting Techniques

- **Answers in one or two lines (1 mark questions)**

1. Define searching.
2. Define sorting.
3. What is Linear Search?
4. What is Binary Search?
5. What is Indexed Sequential Search?
6. What is Bubble Sort?
7. What is Selection Sort?
8. What is Insertion Sort?
9. What is Merge Sort?
10. What is Quick Sort?
11. What is Radix Sort?
12. Which search technique requires sorted data?
13. Which sorting algorithm compares adjacent elements?
14. What is the best-case time complexity of Linear Search?
15. What is the worst-case time complexity of Binary Search?
16. Which sorting algorithm follows the Divide and Conquer technique?
17. Which search algorithm is faster for sorted data?
18. What is a key (search key)?
19. What is swapping in sorting?
20. Which sorting technique repeatedly selects the smallest element?

- **Short Answer Questions (3 marks questions)**

1. Explain the concept of searching.
2. Explain Linear Search with an example.
3. Explain Binary Search with an example.
4. Differentiate between Linear Search and Binary Search.
5. Explain Indexed Sequential Search.
6. Explain the concept of sorting.
7. Explain Bubble Sort with an example.
8. Explain Selection Sort with an example.
9. Explain Insertion Sort with an example.
10. Explain Merge Sort.
11. Explain Quick Sort.
12. Explain Radix Sort.
13. Differentiate between Bubble Sort and Selection Sort.
14. Differentiate between Merge Sort and Quick Sort.
15. Explain the importance of searching and sorting techniques.

- **Long Answer Questions (6 marks questions)**

1. Explain Linear Search and Binary Search with suitable algorithms and examples.
2. Differentiate between Linear Search, Binary Search, and Indexed Sequential Search with a comparison table.
3. Explain Bubble Sort with a suitable algorithm and example.
4. Explain Selection Sort and Insertion Sort with suitable algorithms.
5. Explain Merge Sort and Quick Sort using the Divide and Conquer technique.
6. Explain Radix Sort with a suitable example.
7. Compare all sorting techniques based on time complexity and performance.
8. Explain different searching techniques with suitable algorithms.
9. Explain different sorting techniques with suitable examples.
10. Write algorithms for Linear Search, Binary Search, Bubble Sort, Selection Sort, Insertion Sort, Merge Sort, and Quick Sort.