Ques 1 Find the maximum sub-array sum for the given elements.
{-2, -1, -3, -4, -1, -2, -1, -5, -4}

-3
5
3
-1

Correct Answer : -1

Ques 2 You are given infinite coins of denominations 1, 3, 4. What is the minimum number of coins required to achieve a sum of 7?

1
2
3
4

Correct Answer : 2

Ques 3 You are given infinite coins of denominations 3, 5, 7. Which of the following sum CANNOT be achieved using these coins?

15
16
17
4

Correct Answer : 4

Ques 4 Find the maximum sub-array sum for the given elements.
{2, -1, 3, -4, 1, -2, -1, 5, -4}

3
5
8
6

Correct Answer : 5

Ques 5 You are given infinite coins of denominations 1, 3, 4. What is the total number of ways in which a sum of 7 can be achieved using these coins if the order of the coins is not important?

4
3
5
6

Correct Answer : 5

Ques 6 You are given infinite coins of denominations 5, 7, 9. Which of the following sum CANNOT be achieved using these coins?

50
21
13
23

Correct Answer : 13

Ques 7 Which graph algorithm is used to find the shortest path in a graph with both positive and negative edge weights (but no negative weight cycles)?

Dijkstra's Algorithm
Bellman-Ford Algorithm
Floyd-Warshall Algorithm
Prim's Algorithm

Correct Answer : Bellman-Ford Algorithm

Ques 8 Which graph traversal algorithm guarantees finding the shortest path in an unweighted graph?

Depth-First Search
Dijkstra's Algorithm
Bellman-Ford Algorithm
Breadth-First Search

Correct Answer : Breadth-First Search

Ques 9 Which graph algorithm is used for finding the shortest path from a source node to all other nodes in a graph with non-negative edge weights?

Kruskal's Algorithm
Dijkstra's Algorithm
Bellman-Ford Algorithm
Floyd-Warshall Algorithm

Correct Answer : Dijkstra's Algorithm

Ques 10 Which of the following is an optimal algorithm for solving the "0/1 Knapsack" problem using dynamic programming?

Depth-First Search
Dijkstra’s Algorithm
Dynamic Programming
Greedy Algorithm

Correct Answer : Dynamic Programming

Ques 11 Given the following Java pseudocode, what is being calculated?
java int fib(int n) { if (n <= 1) return n; int prev = 0, curr = 1; for (int i = 2; i <= n; i++) { int next = prev + curr; prev = curr; curr = next; } return curr; }

Factorial of n
Fibonacci number at position n
Sum of the first n numbers
Prime number at position n

Correct Answer : Fibonacci number at position n

Ques 12 Given the following pseudocode, what is the algorithm solving?
java int fib(int n) { int dp[] = new int[n+1]; dp[0] = 0; dp[1] = 1; for (int i = 2; i <= n; i++) { dp[i] = dp[i-1] + dp[i-2]; } return dp[n]; }

Factorial of n

Fibonacci number at position n
Longest Common Subsequence
Matrix multiplication

Correct Answer : Fibonacci number at position n

Ques 13 Which of the following problems can be solved by dynamic programming?

Sorting an array of numbers
Finding the shortest path in a weighted graph
Finding the largest prime factor of a number
Finding the number of ways to make change for a given amount

Correct Answer : Finding the number of ways to make change for a given amount

Ques 14 Which of the following is true for a bipartite graph?

All vertices are connected to each other.
It can be divided into two sets such that no two vertices within the same set are adjacent.
All vertices are connected in a cycle.
It has only one edge.

Correct Answer : It can be divided into two sets such that no two vertices within the same set are adjacent.

Ques 15 Which of the following is TRUE about the Bellman-Ford algorithm for finding the shortest path in a weighted graph?

It only works with graphs containing non-negative edge weights.
It can detect negative weight cycles in the graph.
It is faster than Dijkstra's algorithm for all types of graphs.
It requires each node to be processed only once.

Correct Answer : It can detect negative weight cycles in the graph.

Ques 16 Which of the following is not a property of a minimum spanning tree (MST)?

It connects all the vertices in the graph.
It has the minimum possible weight sum.

It contains exactly V-1 edges, where V is the number of vertices.
It contains all the edges of the graph.

Correct Answer : It contains all the edges of the graph.

Ques 17 Which of the following is true about the Bellman-Ford algorithm?

It finds the shortest path in a graph with negative edge weights but without negative cycles.
It only works with undirected graphs.
It finds the shortest path from a source to only one vertex.
It uses a greedy approach to find the shortest path.

Correct Answer : It finds the shortest path in a graph with negative edge weights but without negative cycles.

Ques 18 Which of the following statements is TRUE about the Floyd-Warshall algorithm for finding shortest paths in a weighted graph?

It is only applicable to directed graphs without cycles.
It finds the shortest paths between all pairs of nodes and can handle negative edge weights, but no negative cycles.
It only finds the shortest path from a single source to all other nodes.
It is more efficient than Dijkstra's algorithm for all types of graphs.

Correct Answer : It finds the shortest paths between all pairs of nodes and can handle negative edge weights, but no negative cycles.

Ques 19 Which of the following is true for dynamic programming?

It is typically used for problems with overlapping subproblems and optimal substructure.
It guarantees an optimal solution by greedy methods.
It uses recursion without memoization.
It is always slower than brute force.

Correct Answer : It is typically used for problems with overlapping subproblems and optimal substructure.

Ques 20 Which of the following is true about the Floyd-Warshall algorithm?

It is used to find the shortest path between two nodes in a graph.
It is used to find the shortest paths between all pairs of nodes.
It is a greedy algorithm.
It only works with undirected graphs.

Correct Answer : It is used to find the shortest paths between all pairs of nodes.

Ques 21 Which of the following is true about topological sorting?

It can be applied only on directed graphs.
It is used to find the minimum spanning tree of a graph.
It orders vertices such that for every directed edge uv, u appears before v.
It is applicable only on undirected graphs.

Correct Answer : It orders vertices such that for every directed edge uv, u appears before v.

Ques 22 Which algorithm can be used to find the minimum spanning tree of a weighted graph?

Dijkstra's Algorithm
Bellman-Ford Algorithm
Kruskal's Algorithm
Depth-First Search

Correct Answer : Kruskal's Algorithm

Ques 23 What does the following pseudocode implement?
java int matrixChainOrder(int[] p) { int n = p.length - 1; int dp[][] = new int[n][n]; for (int len = 2; len <= n; len++) { for (int i = 0; i < n - len + 1; i++) { int j = i + len - 1; dp[i][j] = Integer.MAX_VALUE; for (int k = i; k < j; k++) { dp[i][j] = Math.min(dp[i][j], dp[i][k] + dp[k+1][j] + p[i]*p[k+1]*p[j+1]); } } } return dp[0][n-1]; }

Matrix chain multiplication problem
Longest Common Subsequence
Fibonacci sequence
Rod cutting problem

Correct Answer : Matrix chain multiplication problem

Ques 24 What does the following pseudocode implement?
java int rodCutting(int[] prices, int n) { int dp[] = new int[n+1]; for (int i = 1; i <= n; i++) { for (int j = 1; j <= i; j++) { dp[i] = Math.max(dp[i], prices[j-1] + dp[i-j]); } } return dp[n]; }

Maximum profit from cutting a rod
Minimum cost to cut the rod
Maximum number of pieces that can be obtained from the rod
Longest rod cut

Correct Answer : Maximum profit from cutting a rod

Ques 25 What does the following pseudocode compute?
java int knapsack(int W, int wt[], int val[], int n) { int dp[][] = new int[n+1][W+1]; for (int i = 1; i <= n; i++) { for (int w = 0; w <= W; w++) { if (wt[i-1] <= w) { dp[i][w] = Math.max(val[i-1] + dp[i-1][w-wt[i-1]], dp[i-1][w]); } else { dp[i][w] = dp[i-1][w]; } } } return dp[n][W]; }

Maximum profit that can be obtained from the knapsack problem
Maximum number of items that can be put in the knapsack
Longest Common Subsequence
Fibonacci sequence at position n

Correct Answer : Maximum profit that can be obtained from the knapsack problem

Ques 26 Which of the following pseudocode computes the maximum sum of a subarray in an array of integers (also known as Kadane's Algorithm)?

Longest Common Subsequence
Longest Increasing Subsequence
Maximum Subarray Sum
0/1 Knapsack

Correct Answer : Maximum Subarray Sum

Ques 27 What is the time complexity of the Dijkstra algorithm when implemented using a priority queue (binary heap)?

O(V^2)
O(E log V)
O(V log E)
O(V + E)

Correct Answer : O(E log V)

Ques 28 Which of the following is the time complexity of the Dijkstra's Algorithm using a priority queue implemented with a binary heap?

O(V^2)
O(E log V)
O(V log E)
O(V + E)

Correct Answer : O(E log V)

Ques 29 What is the space complexity of the "Longest Common Subsequence" problem when solved using dynamic programming with a 2D array?

O(1)
O(mn), where m and n are the lengths of the two strings
O(m)
O(n)

Correct Answer : O(mn), where m and n are the lengths of the two strings

Ques 30 What is the space complexity of the "Longest Common Subsequence" (LCS) problem when solved using dynamic programming with a 2D table?

O(1)
O(n)
O(mn), where m and n are the lengths of the two strings
O(n^2)

Correct Answer : O(mn), where m and n are the lengths of the two strings

Ques 31 What is the time complexity of the following Java pseudocode that solves the "Rod Cutting" problem?

O(n)
O(n^2)
O(n * W)

O(n^3)

Correct Answer : O(n^2)

Ques 32 What is the time complexity of the Floyd-Warshall algorithm for finding all pairs shortest paths in a graph with V vertices?

O(V^3)
O(V^2)
O(V^2 log V)
O(VE)

Correct Answer : O(V^3)

Ques 33 What is the space complexity of the "Floyd-Warshall" algorithm for finding all pairs shortest paths in a graph with V vertices?

O(V^3)
O(V^2)
O(V log V)
O(E * log V)

Correct Answer : O(V^3)

Ques 34 What is the time complexity of the "Bellman-Ford" algorithm for finding the shortest path in a graph with V vertices and E edges?

O(V^2)
O(E log V)
O(VE)
O(V + E)

Correct Answer : O(VE)

Ques 35 Which algorithm is used to find the minimum spanning tree (MST) of a connected, undirected graph with weighted edges?

Bellman-Ford Algorithm
Dijkstra's Algorithm

Prim's Algorithm
Floyd-Warshall Algorithm

Correct Answer : Prim's Algorithm

Ques 36 Which of the following is a valid use case for dynamic programming?

Sorting an array of numbers
Finding the maximum value of a set of independent subproblems
Solving problems with overlapping subproblems and optimal substructure
Traversing a tree in pre-order

Correct Answer : Solving problems with overlapping subproblems and optimal substructure

Ques 37 Which of the following is NOT a valid use case for dynamic programming?

Longest Increasing Subsequence
Rod Cutting Problem
Sorting a list of numbers
Fibonacci Sequence

Correct Answer : Sorting a list of numbers

Ques 38 How is an adjacency list represented in a graph?

A matrix of size VxV
A dictionary of lists
A list of edges
A boolean array

Correct Answer : A dictionary of lists

Ques 39 What is the minimum spanning tree of a graph?

A tree with minimum depth
A tree with minimum weight
A tree with maximum edges
A spanning tree with negative cycles

Correct Answer : A tree with minimum weight

Ques 40 Which of the following real-world problems can BFS solve?

Social network connectivity
Network flow problems
Detecting deadlocks
All of the above

Correct Answer : All of the above

Ques 41 What does Floyd-Warshall algorithm compute?

Single source shortest path
Minimum spanning tree
All pairs shortest paths
Topological order

Correct Answer : All pairs shortest paths

Ques 42 What assumption is made in Dijkstra's algorithm for edge weights?

Negative weights are allowed
All weights must be non-negative
All weights must be integers
Edge weights must form a cycle-free graph

Correct Answer : All weights must be non-negative

Ques 43 Floyd-Warshall algorithm is typically used for:

Single source shortest path
All-pairs shortest path
Minimum spanning tree
Longest path in DAG

Correct Answer : All-pairs shortest path

Ques 44 What is the primary application of the Floyd-Warshall algorithm?

Single source shortest path
All-pairs shortest path
Minimum spanning tree
Longest path in DAG

Correct Answer : All-pairs shortest path

Ques 45 How is an edge list stored in memory?

As a list of (u, v) pairs
As a matrix of size VxV
As a binary tree
As a dictionary of lists

Correct Answer : As a list of (u, v) pairs

Ques 46 How is an adjacency list stored in memory?

As a matrix
As a hash map
As an array of linked lists
As a binary tree

Correct Answer : As an array of linked lists

Ques 47 What is the primary advantage of KMP over brute force pattern matching?

Faster matching
Avoids redundant comparisons
Works with negative weights
Simpler implementation

Correct Answer : Avoids redundant comparisons

Ques 48 What is the key advantage of iterative DP (tabulation) over recursive DP (memorization)?

Uses recursion
Uses additional memory
Avoids stack overflow
Reduces time complexity

Correct Answer : Avoids stack overflow

Ques 49 Which of the following is true about Bellman-Ford and Dijkstra's algorithms for shortest path finding?

Bellman-Ford works only on positive weights, while Dijkstra's works on negative weights
Both work only for graphs with non-negative edge weights
Bellman-Ford can handle negative weights, while Dijkstra's cannot
Both can handle graphs with negative weights

Correct Answer : Bellman-Ford can handle negative weights, while Dijkstra's cannot

Ques 50 Which algorithm is best suited to find the shortest path in a binary maze?

BFS
DFS
Dijkstra
Bellman-Ford

Correct Answer : BFS

Ques 51 Why is BFS preferred over DFS for shortest paths in an unweighted graph?

BFS explores deeper nodes first
BFS uses a priority queue
BFS ensures minimum depth exploration
BFS avoids cycles

Correct Answer : BFS ensures minimum depth exploration

Ques 52 What is the purpose of memorization in dynamic programming?

To store solutions to subproblems
To avoid recalculating solved subproblems
To reduce space complexity
Both 1 and 2

Correct Answer : Both 1 and 2

Ques 53 What is the key difference between Greedy Algorithms and Dynamic Programming?

Greedy algorithms always guarantee optimal solutions
DP considers all possibilities
Greedy doesn't consider future consequences
Both 2 and 3

Correct Answer : Both 2 and 3

Ques 54 Which of the following is/are property/properties of a dynamic programming problem?

Optimal substructure
Overlapping subproblems
Greedy approach
Both optimal substructure and overlapping subproblems

Correct Answer : Both optimal substructure and overlapping subproblems

Ques 55 Which of the following is/are property/properties of a dynamic programming problem?

Optimal substructure
Overlapping subproblems
Greedy approach
Both optimal substructure and overlapping subproblems

Correct Answer : Both optimal substructure and overlapping subproblems

Ques 56 Bellman-Ford algorithm can handle:

Only positive edges
Only negative edges
Both positive and negative edges
Only zero-weight edges

Correct Answer : Both positive and negative edges

Ques 57 In the context of Minimum Spanning Tree (MST), if two edges have the same weight, what will happen?

Prim's algorithm will always choose the first one
Both Prim's and Kruskal's algorithms can choose any of the two edges
Kruskal's algorithm will ignore one of the edges
The graph will no longer have an MST

Correct Answer : Both Prim's and Kruskal's algorithms can choose any of the two edges

Ques 58 What is the correct order to solve a DP problem using tabulation?

Bottom-up approach
Top-down approach
Randomized approach
Brute force

Correct Answer : Bottom-up approach

Ques 59 Which of the following best describes the main idea behind Dynamic Programming (DP) when solving a problem?

Repeating recursive calls for each subproblem without storing results.
Breaking down a problem into overlapping subproblems and storing their solutions to avoid redundant calculations.
Breaking down a problem into completely independent subproblems that don't interact.
Solving each subproblem sequentially without any need for optimization.

Correct Answer : Breaking down a problem into overlapping subproblems and storing their solutions to avoid redundant calculations.

Ques 60 Which of the following best describes the tabulation approach in Dynamic Programming?

Solving subproblems recursively and storing results to avoid duplicate work.
Building a solution iteratively by solving all smaller subproblems first and storing the results in a table.
Solving subproblems in any order and backtracking to find the final answer.
Creating a table and filling it with pre-computed results before starting the main algorithm.

Correct Answer : Building a solution iteratively by solving all smaller subproblems first and storing the results in a table.

Ques 61 How does Bellman-Ford algorithm detect a negative weight cycle?

By checking negative weights in the graph
By comparing weights after V iterations
By checking for relaxed edges in the V-th iteration
By calculating the minimum spanning tree

Correct Answer : By checking for relaxed edges in the V-th iteration

Ques 62 How does Floyd-Warshall detect negative weight cycles?

By checking the main diagonal of the matrix
By checking for unvisited vertices
By examining shortest paths
By running DFS after all iterations

Correct Answer : By checking the main diagonal of the matrix

Ques 63 How does the Floyd-Warshall algorithm detect negative weight cycles?

By detecting zero-weight edges
By comparing distances
By looking at the diagonal of the distance matrix
By checking edge weights

Correct Answer : By looking at the diagonal of the distance matrix

Ques 64 In dynamic programming, what does memoization refer to?

Storing intermediate results
Iterative calculation
Recursive division
Caching subproblem solutions

Correct Answer : Caching subproblem solutions

Ques 65 How does Floyd-Warshall detect negative weight cycles?

Detects zero-weight edges
Compares diagonal values
Looks for negative distances
Checks adjacency matrix

Correct Answer : Compares diagonal values

Ques 66 When a top-down approach of dynamic programming is applied to a problem, it usually _____

Decreases both, the time complexity and the space complexity
Decreases the time complexity and increases the space complexity
Increases the time complexity and decreases the space complexity
Increases both, the time complexity and the space complexity

Correct Answer : Decreases the time complexity and increases the space complexity

Ques 67 Which of the following problems is not typically solved by dynamic programming?

Longest Increasing Subsequence
Matrix Chain Multiplication
Depth-First Search
0/1 Knapsack Problem

Correct Answer : Depth-First Search

Ques 68 Which graph traversal algorithm is used for topological sorting?

BFS
DFS
Dijkstra
Prim

Correct Answer : DFS

Ques 69 Which algorithm is preferred for shortest path in weighted graphs with non-negative weights?

BFS
Bellman-Ford
Dijkstra
Floyd-Warshall

Correct Answer : Dijkstra

Ques 70 Which of the following algorithms is not used to find an MST?

Dijkstra's Algorithm
Prim's Algorithm
Kruskal's Algorithm
Borůvka's Algorithm

Correct Answer : Dijkstra's Algorithm

Ques 71 Which data structure is essential for implementing Kruskal's algorithm efficiently?

Stack
Queue
Disjoint Set (Union-Find)
Heap

Correct Answer : Disjoint Set (Union-Find)

Ques 72 Which data structure is used to detect cycles in Kruskal's algorithm?

Queue

Disjoint Sets
Priority Queue
Stack

Correct Answer : Disjoint Sets

Ques 73 If a problem can be solved by combining optimal solutions to non-overlapping problems, the strategy is called _____

Dynamic programming
Greedy
Divide and conquer
Recursion

Correct Answer : Divide and conquer

Ques 74 In the Longest Common Subsequence problem, which of the following approaches is generally used?

Divide and conquer
Dynamic Programming
Breadth-First Search
Recursion

Correct Answer : Dynamic Programming

Ques 75 The Fibonacci sequence is a classic example of a problem that can be solved using:

Greedy algorithms
Dynamic programming
Backtracking
Branch and bound

Correct Answer : Dynamic programming

Ques 76 The 0/1 Knapsack problem is best solved using which technique?

Greedy Algorithms
Divide and Conquer

Dynamic Programming
Backtracking

Correct Answer : Dynamic Programming

Ques 77 Which technique is most suitable for solving the 0/1 Knapsack problem?

Greedy Algorithms
Divide and Conquer
Dynamic Programming
Backtracking

Correct Answer : Dynamic Programming

Ques 78 The Fibonacci sequence is an example of a problem solved using:

Dynamic programming
Greedy algorithms
Backtracking
Divide and Conquer

Correct Answer : Dynamic programming

Ques 79 Given a one-dimensional array of integers, you have to find a sub-array with maximum sum. This is the maximum sub-array sum problem. Which of these methods can be used to solve the problem?

Dynamic programming
Two for loops (naive method)
Divide and conquer
Dynamic programming, naïve method and Divide and conquer methods

Correct Answer : Dynamic programming, naïve method and Divide and conquer methods

Ques 80 In the Z-value algorithm, which of the following statements correctly describes the purpose of the Z-array for a given string?

Each entry in the Z-array stores the number of times the string's characters appear in alphabetical order.

Each entry
$Z[i]$ in the Z-array stores the length of the longest substring starting from i that matches a prefix of the string.
The Z-array values increase for positions where the characters are lexicographically larger than the first character of the string.
The Z-array is only used for strings that are palindromes.

Correct Answer : Each entry
$Z[i]$ in the Z-array stores the length of the longest substring starting from i that matches a prefix of the string.

Ques 81 Which graph representation is more suitable for sparse graphs?

Adjacency matrix
Edge list
Adjacency list
Distance matrix

Correct Answer : Edge list

Ques 82 What is the difference between an edge list and an adjacency matrix?

Edge list uses less space than adjacency matrix
Adjacency matrix is more efficient in searches
Edge list supports weighted graphs
Both 1 and 2

Correct Answer : Edge list uses less space than adjacency matrix

Ques 83 What is the characteristic of BFS traversal?

Explores nodes layer by layer
Explores deeper nodes first
Avoids cycles
Uses a stack

Correct Answer : Explores nodes layer by layer

Ques 84 What is the time complexity of the recursive implementation used to find the nth fibonacci term?

O(1)
O(n2)
O(n!)
Exponential

Correct Answer : Exponential

Ques 85 Consider the recursive implementation to find the nth fibonacci number:

```
int fibo(int n)
   if n <= 1
         return n
   return _____
```

Which line would make the implementation complete?

fibo(n) + fibo(n)
fibo(n) + fibo(n – 1)
fibo(n – 1) + fibo(n + 1)
fibo(n – 1) + fibo(n – 2)

Correct Answer : fibo(n – 1) + fibo(n – 2)

Ques 86 Which of these is a classic example of an optimization problem solved using dynamic programming?

Fibonacci sequence
Prim's Algorithm
Shortest Path
Tower of Hanoi

Correct Answer : Fibonacci sequence

Ques 87 What is the primary application of Floyd-Warshall algorithm?

Finding MST
Finding all-pairs shortest paths
Detecting negative cycles

Determining graph connectivity

Correct Answer : Finding all-pairs shortest paths

Ques 88 Which of the following problems is NOT solved using dynamic programming?

0/1 knapsack problem
Matrix chain multiplication problem
Edit distance problem
Fractional knapsack problem

Correct Answer : Fractional knapsack problem

Ques 89 Kruskal's algorithm is based on which approach?

Greedy approach
Dynamic programming
Backtracking
Divide and Conquer

Correct Answer : Greedy approach

Ques 90 What is the main advantage of Bellman-Ford over Dijkstra's algorithm?

Handles negative weights
Faster
Simpler
Uses less memory

Correct Answer : Handles negative weights

Ques 91 When is an adjacency matrix more efficient than an adjacency list?

In sparse graphs
In dense graphs
In unweighted graphs
In directed graphs

Correct Answer : In dense graphs

Ques 92 In Kruskal's algorithm, how are edges sorted?

In decreasing order of weights
In increasing order of weights
Randomly
By the degree of endpoints

Correct Answer : In increasing order of weights

Ques 93 Which of the following statements is TRUE about Prim's Algorithm for finding a Minimum Spanning Tree (MST) in a connected, weighted graph?

It always starts with the edge of the minimum weight in the graph.
It grows the MST by selecting the smallest edge that connects a node in the MST to a node outside of it.
It only works with graphs that have unique edge weights.
It finds the shortest path between all pairs of nodes in the graph.

Correct Answer : It grows the MST by selecting the smallest edge that connects a node in the MST to a node outside of it.

Ques 94 What happens when a node is visited in DFS traversal?

It is marked as visited
It is pushed into the queue
It is revisited in cycles
All its neighbors are explored simultaneously

Correct Answer : It is marked as visited

Ques 95 How does Bellman-Ford handle negative weights?

It uses backtracking
It iteratively relaxes edges V-1 times
It calculates all-pairs shortest paths
It discards negative cycles

Correct Answer : It iteratively relaxes edges V-1 times

Ques 96 Which of the following statements is TRUE regarding Manacher's Algorithm for finding the longest palindromic substring in a string?

It works in
$O(n$^2) time by expanding around every possible center in the string.
It preprocesses the string by adding special characters to handle even-length palindromes and achieves
$O(n)$ time complexity.
It can only find palindromic substrings of odd length.
The algorithm requires dynamic programming to store intermediate results.

Correct Answer : It preprocesses the string by adding special characters to handle even-length palindromes and achieves
$O(n)$ time complexity.

Ques 97 Which of the following statements is TRUE regarding topological sorting in a Directed Acyclic Graph (DAG)?

Topological sorting can only be applied to undirected graphs.
It provides a linear ordering of nodes such that for every directed edge
$u \rightarrow v$, u appears before v in the ordering.
Topological sorting works correctly even if the graph has cycles.
There is only one unique topological order for any given DAG.

Correct Answer : It provides a linear ordering of nodes such that for every directed edge
$u \rightarrow v$, u appears before v in the ordering.

Ques 98 In Dijkstra's algorithm for finding the shortest path in a graph with non-negative edge weights, which of the following statements is TRUE about how it selects the next node to process?

It selects the node with the maximum edge weight connected to the start node.
It always processes nodes in the order they were added to the graph.
It selects the unvisited node with the smallest known distance from the starting node.
It selects nodes randomly and updates distances accordingly.

Correct Answer : It selects the unvisited node with the smallest known distance from the starting node.

Ques 99 Why is Dijkstra's algorithm efficient for shortest paths in weighted graphs?

It only works for undirected graphs
It uses a priority queue
It can handle negative weights
It relaxes edges at each iteration

Correct Answer : It uses a priority queue

Ques 100 Why is Bellman-Ford preferred over Dijkstra in certain cases?

It is faster
It works with graphs having cycles
It works with negative weights
It does not use edge relaxation

Correct Answer : It works with negative weights

Ques 101 Which algorithm is efficient for pattern matching with partial matches?

Naive Algorithm
KMP Algorithm
Binary Search
Z-Algorithm

Correct Answer : KMP Algorithm

Ques 102 Which of the following problems is typically solved using dynamic programming?

Knapsack
Sorting
Binary Search
Maximum Flow

Correct Answer : Knapsack

Ques 103 Which dynamic programming problem aims to minimize the sum of selected weights without exceeding a limit?

Fibonacci sequence
Subset Sum
Knapsack Problem
Minimum Spanning Tree

Correct Answer : Knapsack Problem

Ques 104 What does the Z-value array store in the Z-algorithm?

Lengths of suffixes matching the prefix of the string
Indices of matching substrings
Number of distinct substrings
None of the above

Correct Answer : Lengths of suffixes matching the prefix of the string

Ques 105 Which of the following problems should be solved using dynamic programming?

Mergesort
Binary search
Longest common subsequence
Quicksort

Correct Answer : Longest common subsequence

Ques 106 Which of the following problems is typically solved using dynamic programming due to subproblem overlap?

Longest common subsequence
Binary search
Quick Sort
Depth-first search

Correct Answer : Longest common subsequence

Ques 107 What does the LPS array in KMP algorithm represent?

Longest Prefix Suffix
Longest Pattern Substring
Longest Prefix Match
None of the above

Correct Answer : Longest Prefix Suffix

Ques 108 Which of the following is NOT a valid representation of graphs?

Edge List
Adjacency Matrix
Adjacency List
Matrix Tree Representation

Correct Answer : Matrix Tree Representation

Ques 109 Which of the following is not a valid way to represent a graph?

Edge List
Adjacency Matrix
Adjacency List
Matrix Tree Representation

Correct Answer : Matrix Tree Representation

Ques 110 In dynamic programming, the technique of storing the previously calculated values is called _____

Saving value property
Storing value property
Memoization
Mapping

Correct Answer : Memoization

Ques 111 Consider the following code to find the nth fibonacci term using dynamic programming:

```
1. int fibo(int n)
2.         int fibo_terms[100000]  //arr to store the fibonacci numbers
3.         fibo_terms[0] = 0
4.         fibo_terms[1] = 1
5.
6.         for i: 2 to n
7.                   fibo_terms[i] = fibo_terms[i - 1] + fibo_terms[i - 2]
8.
9.         return fibo_terms[n]
```

Which technique is used by line 7 of the above code?

Optimal substructure
Overlapping subproblems
Memoization
Greedy substructure

Correct Answer : Memoization

Ques 112 Which of the following is true about a Minimum Spanning Tree (MST)?

MST always has the same weight as the shortest path tree
MST always includes all vertices of the graph
MST is unique for every graph
MST must be a directed acyclic graph

Correct Answer : MST always includes all vertices of the graph

Ques 113 In a complete graph with n vertices, how many edges are there in its minimum spanning tree?

n
n - 1
n + 1
2n

Correct Answer : n - 1

Ques 114 In a Depth-First Search (DFS) traversal of a graph, which of the following statements is TRUE about the order in which nodes are visited?

Nodes closer to the starting node are always visited first.
Nodes are visited in a recursive manner, going as deep as possible along each path before backtracking.
Nodes with the highest degree are visited first.
Nodes are visited in the reverse order of their addition to the graph.

Correct Answer : Nodes are visited in a recursive manner, going as deep as possible along each path before backtracking.

Ques 115 When performing a Breadth-First Search (BFS) on an undirected graph from a given starting node, which of the following is TRUE about the order in which nodes are visited?

Nodes are visited in the order they were added to the graph.
Nodes closer to the starting node in terms of edge distance are visited earlier.
Nodes are visited based on their degree, with higher degree nodes visited first.
Nodes are visited randomly regardless of their distance from the starting node.

Correct Answer : Nodes closer to the starting node in terms of edge distance are visited earlier.

Ques 116 If a graph has negative edge weights, which algorithm cannot be used directly?

Prim's Algorithm
Kruskal's Algorithm
Borůvka's Algorithm
None of the above

Correct Answer : None of the above

Ques 117 What is the time complexity of Dijkstra's algorithm using a binary heap and adjacency list?

O(V + E)
O((V + E) * log V)
O(V^2)
O(V^3)

Correct Answer : O((V + E) * log V)

Ques 118 What is the space complexity of the recursive implementation used to find the nth fibonacci term?

O(1)
O(n)
On^2)
O(n^3)

Correct Answer : O(1)

Ques 119 What is the space complexity of the divide and conquer algorithm used to find the maximum sub-array sum?

O(n)
O(1)
O(n!)
O(n2)

Correct Answer : O(1)

Ques 120 What is the space complexity of the following for loop method used to compute the nth fibonacci term?

```
int fibo(int n)
   if n == 0
      return 0
   else
      prevFib = 0
      curFib = 1
      for i : 1 to n-1
         nextFib = prevFib + curFib
             prevFib = curFib
         curFib = nextFib
      return curFib
```

O(1)
O(n)
O(n2)
Exponential

Correct Answer : O(1)

Ques 121 What is the time complexity to find all neighbors of a node in an adjacency list?

O(1)
O(V)
O(E)
O(degree of the node)

Correct Answer : O(degree of the node)

Ques 122 What is the time complexity of Dijkstra's Algorithm with a priority queue?

O(V^2)
O(E log V)
O(V log E)
O(E^2)

Correct Answer : O(E log V)

Ques 123 What is the time complexity of Kruskal's algorithm using a union-find data structure?

O(E log V)
O(V log V)
O(E log E)
O(V^2)

Correct Answer : O(E log V)

Ques 124 The worst-case time complexity of the Knuth-Morris-Pratt (KMP) algorithm is:

O(n^2)
O(n log n)
O(n)
O(n + m)

Correct Answer : O(n + m)

Ques 125 Which of the following best describes the KMP algorithm's behavior on a pattern with all unique characters?

O(n^2)
O(m) only
O(n + m)
None of the above

Correct Answer : O(n + m)

Ques 126 What is the time complexity of KMP string matching algorithm?

O(n^2)
O(n log n)
O(n + m)
O(nm)

Correct Answer : O(n + m)

Ques 127 What is the time complexity of the KMP string matching algorithm?

O(n^2)
O(n log n)
O(n + m)
O(nm)

Correct Answer : O(n + m)

Ques 128 In the Z-algorithm, what is the time complexity of computing the Z-values for a string of length n?

O(n^2)
O(n log n)
O(n)
O(log n)

Correct Answer : O(n)

Ques 129 What is the time complexity of the following dynamic programming implementation used to compute the nth fibonacci term?

```
1. int fibo(int n)
2.       int fibo_terms[100000]  //arr to store the fibonacci numbers
3.       fibo_terms[0] = 0
4.       fibo_terms[1] = 1
5.
6.       for i: 2 to n
7.                fibo_terms[i] = fibo_terms[i - 1] + fibo_terms[i - 2]
8.
9.       return fibo_terms[n]
```

O(1)
O(n)
On^2)
O(n^3)

Correct Answer : O(n)

Ques 130 What is the space complexity of the following dynamic programming implementation used to compute the nth fibonacci term?

```
int fibo(int n)
      int fibo_terms[100000]  //arr to store the fibonacci numbers
      fibo_terms[0] = 0
         fibo_terms[1] = 1

         for i: 2 to n
                  fibo_terms[i] = fibo_terms[i - 1] + fibo_terms[i - 2]

         return fibo_terms[n]
```

O(1)
O(n)
On^2)
O(n^3)

Correct Answer : O(n)

Ques 131 What is the time complexity of the following for loop method used to compute the nth fibonacci term?

```
int fibo(int n)
   if n == 0
     return 0
   else
     prevFib = 0
     curFib = 1
     for i : 1 to n-1
        nextFib = prevFib + curFib
            prevFib = curFib
        curFib = nextFib
     return curFib
```

O(1)
O(n)
O(n2)
Exponential

Correct Answer : O(n)

Ques 132 What is the time complexity of the following Java pseudocode that solves the "Longest Common Subsequence" problem?
```java int lcs(String X, String Y) { int m = X.length(); int n = Y.length(); int dp[][] = new int[m+1][n+1]; for (int i = 0; i <= m; i++) { for (int j = 0; j <= n; j++) { if (i == 0

o(n)
j == 0) { dp[i][j] = 0; } else if (X.charAt(i-1) == Y.charAt(j-1)) { dp[i][j] = dp[i-1][j-1] + 1; } else { dp[i][j] = Math.max(dp[i-1][j], dp[i][j-1]); } } } return dp[m][n]; }```
O(n^2)
O(n * m), where n and m are the lengths of the two strings

Correct Answer : o(n)

Ques 133 What is the time complexity of solving Fibonacci using tabulation?

O(n)
O(log n)
O(2^n)
O(1)

Correct Answer : O(n)
```

Ques 134 What is the space complexity of an adjacency matrix in a graph with 'n' vertices?

O(n)
O(n^2)
O(n log n)
O(2^n)

Correct Answer : O(n^2)

Ques 135 What is the space complexity of an adjacency matrix for a graph with n vertices?

O(n)
O(n^2)
O(n log n)
O(2^n)

Correct Answer : O(n^2)

Ques 136 What is the space complexity of an adjacency matrix for a graph with n vertices?

O(n)
O(n^2)
O(n log n)
O(n^3)

Correct Answer : O(n^2)

Ques 137 What is the time complexity of the divide and conquer algorithm used to find the maximum sub-array sum?

O(n)
O(logn)
O(nlogn)
O(n2)

Correct Answer : O(nlogn)

Ques 138 What is the time complexity of solving the 0/1 Knapsack problem using dynamic programming?

O(n)
O(nW)
O(2^n)
O(log n)

Correct Answer : O(nW)

Ques 139 What is the time complexity of the Bellman-Ford algorithm for V vertices and E edges?

O(V^2)
O(V * E)
O(V + E)
O(V^3)

Correct Answer : O(V * E)

Ques 140 What is the space complexity of an adjacency list for a graph with V vertices and E edges?

O(V + E)
O(V^2)
O(V log E)
O(VE)

Correct Answer : O(V + E)

Ques 141 What is the worst-case time complexity of BFS on a graph with V vertices and E edges?

O(V)
O(V + E)
O(E^2)
O(V^2)

Correct Answer : O(V + E)

Ques 142 What is the space complexity of the Floyd-Warshall algorithm for a graph with V vertices?

O(V)
O(V^2)
O(V + E)
O(V^3)

Correct Answer : O(V^2)

Ques 143 What is the time complexity of Prim's algorithm using an adjacency matrix for V vertices?

O(V^2)
O(V + E)
O((V + E) * log V)
O(V^3)

Correct Answer : O(V^2)

Ques 144 What is the space complexity of an adjacency matrix for a graph with V vertices?

O(V)
O(V^2)
O(E)
O(V + E)

Correct Answer : O(V^2)

Ques 145 What is the time complexity of Floyd-Warshall Algorithm for a graph with V vertices?

O(VE)
O(V^2)
O(V^3)
O(E log V)

Correct Answer : O(V^3)

Ques 146 What is the time complexity of the Floyd-Warshall algorithm for a graph with 'V' vertices?

O(VE)
O(V^2)
O(V^3)
O(E log V)

Correct Answer : O(V^3)

Ques 147 What is the time complexity of Floyd-Warshall algorithm for a graph with V vertices?

O(VE)
O(V^2)
O(V^3)
O(E log V)

Correct Answer : O(V^3)

Ques 148 What is the space complexity of an adjacency list for a graph with V vertices and E edges?

O(V^2)
O(V+E)
O(E log V)
O(V+E log V)

Correct Answer : O(V+E)

Ques 149 What is the space complexity of an adjacency list in a graph with V vertices and E edges?

O(V^2)
O(V+E)
O(E log V)
O(V+E log V)

Correct Answer : O(V+E)

Ques 150 What is the time complexity of the Bellman-Ford algorithm for a graph with 'V' vertices and 'E' edges?

O(V+E)
O(VE)
O(E log V)
O(V^2)

Correct Answer : O(VE)

Ques 151 What is the time complexity of Bellman-Ford algorithm in a graph with V vertices and E edges?

O(V+E)
O(VE)
O(E log V)
O(V^2)

Correct Answer : O(VE)

Ques 152 What is the time complexity of the Bellman-Ford algorithm for a graph with V vertices and E edges?

O(VE)
O(V^2)
O(E log V)
O(V + E)

Correct Answer : O(VE)

Ques 153 If an optimal solution can be created for a problem by constructing optimal solutions for its subproblems, the problem possesses _____ property.

Overlapping subproblems
Optimal substructure
Memoization
Greedy

Correct Answer : Optimal substructure

Ques 154 Consider the following code to find the nth fibonacci term using dynamic programming:

```
1. int fibo(int n)
2.   int fibo_terms[100000]  //arr to store the fibonacci numbers
3.   fibo_terms[0] = 0
4.   fibo_terms[1] = 1
5.
6.   for i: 2 to n
7.         fibo_terms[i] = fibo_terms[i - 1] + fibo_terms[i - 2]
8.
9.   return fibo_terms[n]
```

Which property is shown by line 7 of the above code?

Optimal substructure
Overlapping subproblems
Both overlapping subproblems and optimal substructure
Greedy substructure

Correct Answer : Optimal substructure

Ques 155 Consider the following code snippet. Which property is shown by line 4 of the below code snippet?

```
1. int sum[len], idx;
2. sum[0] = arr[0];
3. for(idx = 1; idx < len; idx++)
4.          sum[idx] = max(sum[idx - 1] + arr[idx], arr[idx]);
5. int mx = sum[0];
6. for(idx = 0; idx < len; idx++)
7.         if(sum[idx] > mx)
8.                  mx =sum[idx];
9. return mx;
```

Optimal substructure
Overlapping subproblems
Memoization
Greedy substructure

Correct Answer : Optimal substructure

Ques 156 In dynamic programming, which property ensures that the solution to a problem can be constructed efficiently from solutions of smaller subproblems?

Greedy choice property
Overlapping subproblems
Optimal substructure
Divide and conquer

Correct Answer : Optimal substructure

Ques 157 What is the key property that allows dynamic programming solutions to work efficiently?

Optimal Substructure
Greedy Choice Property
No overlapping subproblems
Independent decisions

Correct Answer : Optimal Substructure

Ques 158 What key feature makes dynamic programming algorithms perform efficiently?

Optimal Substructure
Greedy Choice Property
No overlapping subproblems
Independent decisions

Correct Answer : Optimal Substructure

Ques 159 If an optimal solution can be created for a problem by constructing optimal solutions for its subproblems, the problem possesses _____ property.

Overlapping subproblems
Optimal substructure
Memoization
Greedy

Correct Answer : Optimal substructure

Ques 160 What is the core idea behind dynamic programming?

Divide and Conquer
Recursive Backtracking
Optimal Substructure and Overlapping Subproblems
Memoization and Tabulation

Correct Answer : Optimal Substructure and Overlapping Subproblems

Ques 161 If a problem can be broken into subproblems which are reused several times, the problem possesses _____ property.

Overlapping subproblems
Optimal substructure
Memoization
Greedy

Correct Answer : Overlapping subproblems

Ques 162 Suppose we find the 8th term using the recursive implementation. The arguments passed to the function calls will be as follows:

fibonacci(8)
fibonacci(7) + fibonacci(6)
fibonacci(6) + fibonacci(5) + fibonacci(5) + fibonacci(4)
fibonacci(5) + fibonacci(4) + fibonacci(4) + fibonacci(3) + fibonacci(4)
+ fibonacci(3) + fibonacci(3) + fibonacci(2)
:
:
:
Which property is shown by the above function calls?

Overlapping subproblems
Optimal substructure
Memoization
Greedy

Correct Answer : Overlapping subproblems

Ques 163 Which two properties are essential for applying dynamic programming?

Greedy and Divide and Conquer
Overlapping Subproblems and Optimal Substructure
Optimal Substructure and Greedy
Divide and Conquer and Memoization

Correct Answer : Overlapping Subproblems and Optimal Substructure

Ques 164 Which array is essential in the KMP algorithm?

Z-array
Prefix-suffix (LPS) array
Suffix array
None of the above

Correct Answer : Prefix-suffix (LPS) array

Ques 165 Consider the following code to find the nth fibonacci term:

```
int fibo(int n)
   if n == 0
     return 0
   else
     prevFib = 0
     curFib = 1
     for i : 1 to n-1
        nextFib = prevFib + curFib
            __________
            __________
     return curFib
```

Complete the above code.

prevFib = curFib
curFib = curFib
prevFib = nextFib
curFib = prevFib
prevFib = curFib
curFib = nextFib
prevFib = nextFib

nextFib = curFib

Correct Answer : prevFib = curFib
curFib = nextFib

Ques 166 What is the primary difference between Prim's and Kruskal's algorithms for finding a minimum spanning tree?

Prim's works only on dense graphs
Kruskal's requires the graph to be directed
Prim's builds the tree one node at a time, while Kruskal's builds it by edges
None of the above

Correct Answer : Prim's builds the tree one node at a time, while Kruskal's builds it by edges

Ques 167 What is the primary difference between Prim's and Kruskal's algorithms?

Prim's uses adjacency list
Kruskal's uses adjacency matrix
Prim's grows a tree
Kruskal's grows a forest

Correct Answer : Prim's grows a tree

Ques 168 In Prim's algorithm, which data structure is often used to select the minimum weight edge?

Disjoint Set
Priority Queue
Stack
Queue

Correct Answer : Priority Queue

Ques 169 What core feature do problems suitable for dynamic programming share?

Problems with overlapping subproblems
Problems with no dependencies
Problems with simple recursion

Problems with greedy choices

Correct Answer : Problems with overlapping subproblems

Ques 170 What is the key characteristic of problems solved by dynamic programming?

Problems with overlapping subproblems
Problems with no dependencies
Problems with simple recursion
Problems with greedy choices

Correct Answer : Problems with overlapping subproblems

Ques 171 Which data structure is commonly used for BFS traversal?

Stack
Queue
Priority Queue
Set

Correct Answer : Queue

Ques 172 The following sequence is a fibonacci sequence:
0, 1, 1, 2, 3, 5, 8, 13, 21,.....
Which technique can be used to get the nth fibonacci term?

Recursion
Dynamic programming
A single for loop
Recursion, Dynamic Programming, For loops

Correct Answer : Recursion, Dynamic Programming, For loops

Ques 173 What is the key advantage of the KMP algorithm over brute force pattern matching?

Reduces redundant comparisons
Faster on average
Uses less space

Works with negative weights

Correct Answer : Reduces redundant comparisons

Ques 174 What is a real-world application of Minimum Spanning Trees?

Routing network connections
Detecting deadlocks
Solving shortest paths
Pathfinding in mazes

Correct Answer : Routing network connections

Ques 175 Which of the following cannot be solved using BFS in a binary maze?

Finding shortest path
Counting number of paths
Detecting unreachable areas
Solving weighted path problems

Correct Answer : Solving weighted path problems

Ques 176 What is the key preprocessing step in Kruskal's algorithm?

Relaxing edges
Sorting edges by weight
Traversing vertices
Calculating shortest path

Correct Answer : Sorting edges by weight

Ques 177 Which data structure is commonly used for DFS traversal?

Stack
Queue
Priority Queue
Hash Map

Correct Answer : Stack

Ques 178 What is the key concept of Prim's algorithm for finding MST?

Start from an arbitrary vertex
Select minimum weight edge
Relax all edges at once
Traverse all vertices

Correct Answer : Start from an arbitrary vertex

Ques 179 In the KMP algorithm, what is the purpose of the prefix table (pi-table)?

Track mismatches
Record shifts in pattern
Store pattern frequencies
Store longest prefix-suffix match

Correct Answer : Store longest prefix-suffix match

Ques 180 In the KMP algorithm, the purpose of the LPS array is to:

Store longest suffixes of substrings
Store the longest prefix which is also a suffix
Track visited indices
Match the pattern in reverse order

Correct Answer : Store the longest prefix which is also a suffix

Ques 181 Which approach to solving the Fibonacci problem uses tabulation?

Recursion
Storing solutions in a table iteratively
Backtracking
Greedy approach

Correct Answer : Storing solutions in a table iteratively

Ques 182 In dynamic programming, what is “memoization”?

Dividing the problem into subproblems
Storing solutions of subproblems to avoid recomputation
A technique to optimize graph traversal
None of the above

Correct Answer : Storing solutions of subproblems to avoid recomputation

Ques 183 What is the core feature of problems solved by dynamic programming?

Greedy property
Subproblem overlap
Divide and conquer
Random selection

Correct Answer : Subproblem overlap

Ques 184 For an undirected graph, how is the adjacency matrix represented?

Symmetrical
Asymmetrical
Diagonal
Random

Correct Answer : Symmetrical

Ques 185 How is the adjacency matrix represented for an undirected graph?

Symmetrical
Asymmetrical
Diagonal
Random

Correct Answer : Symmetrical

Ques 186 How does tabulation differ from memorization?

Tabulation uses recursion
Tabulation solves subproblems iteratively
Tabulation uses additional memory
Tabulation does not require problem overlap

Correct Answer : Tabulation solves subproblems iteratively

Ques 187 In Kruskal's Algorithm for finding a Minimum Spanning Tree (MST), which of the following is TRUE about how edges are processed to ensure that cycles are avoided in the MST?

Edges are added to the MST in the order they appear in the graph's adjacency list.
The algorithm uses a union-find (disjoint-set) data structure to detect and prevent cycles by tracking connected components of nodes.
It relies on a breadth-first search (BFS) to avoid cycles while adding edges.
It only works on graphs with unique edge weights to ensure no cycles are formed.

Correct Answer : The algorithm uses a union-find (disjoint-set) data structure to detect and prevent cycles by tracking connected components of nodes.

Ques 188 What is the condition to move to the next cell in a binary maze shortest path problem?

The cell is visited
The cell is unvisited and open
The cell is visited and closed
The cell is blocked

Correct Answer : The cell is unvisited and open

Ques 189 What does the priority queue store in Dijkstra's algorithm?

Only the source vertex
The current shortest distances
All edges
All vertices

Correct Answer : The current shortest distances

Ques 190 What happens when you relax an edge in the Bellman-Ford algorithm?

The edge is removed
The vertex is marked as visited
The distance to the vertex is updated
The edge weight is modified

Correct Answer : The distance to the vertex is updated

Ques 191 In a Directed Acyclic Graph (DAG), which of the following statements about topological sorting is CORRECT in terms of the number of possible topological orders?

A unique topological order exists for any DAG
The number of possible topological orderings is always equal to the number of vertices in the DAG.
The number of topological orderings depends on the presence of vertices with no incoming or outgoing edges.
The number of possible topological orderings depends on the structure of the DAG, with more than one order possible when multiple vertices share dependencies.

Correct Answer : The number of possible topological orderings depends on the structure of the DAG, with more than one order possible when multiple vertices share dependencies.

Ques 192 In the KMP (Knuth-Morris-Pratt) algorithm, which of the following statements is TRUE about the construction and use of the prefix function (or LPS array)?

The prefix function records the longest proper prefix of the pattern that is also a suffix, helping avoid unnecessary comparisons.
The prefix function stores the number of occurrences of each character in the pattern.
The prefix function is recalculated from scratch each time a mismatch occurs.
The prefix function is only needed if the pattern contains repeating characters.

Correct Answer : The prefix function records the longest proper prefix of the pattern that is also a suffix, helping avoid unnecessary comparisons.

Ques 193 In Dynamic Programming, what is the primary purpose of using memoization?

To avoid recalculating solutions to the same subproblems by storing their results.
To ensure that all subproblems are recalculated every time they are needed.
To improve space complexity by storing all results in an array.
To break down problems into entirely independent subproblems.

Correct Answer : To avoid recalculating solutions to the same subproblems by storing their results.

Ques 194 What is the purpose of a queue in BFS?

To explore deeper levels
To maintain the order of node traversal
To detect cycles
To optimize space usage

Correct Answer : To maintain the order of node traversal

Ques 195 What is the purpose of the prefix table (pi-table) in the KMP algorithm?

To store the matches found
To reduce redundant comparisons
To count the number of occurrences
To avoid backtracking

Correct Answer : To reduce redundant comparisons

Ques 196 What role does the prefix table (pi-table) play in the KMP algorithm?

To store the matches found
To reduce redundant comparisons
To count the number of occurrences
To avoid backtracking

Correct Answer : To reduce redundant comparisons

Ques 197 Which of the following is NOT a valid application of BFS?

Shortest path in unweighted graph
Checking bipartiteness
Topological sorting
Finding connected components

Correct Answer : Topological sorting

Ques 198 What does a Minimum Spanning Tree (MST) minimize?

Number of vertices
Number of edges
Total edge weight
Maximum degree of vertices

Correct Answer : Total edge weight

Ques 199 What is the role of a disjoint set in Kruskal's algorithm?

Sort edges by weight
Track connected components
Find minimum edge
Avoid negative cycles

Correct Answer : Track connected components

Ques 200 Which of the following problems can be solved by applying dynamic programming with overlapping subproblems and optimal substructure properties?

Traveling Salesman Problem
Depth-First Search
Breadth-First Search
Cycle Detection in Graphs

Correct Answer : Traveling Salesman Problem

Ques 201 What data structure is essential for implementing Kruskal's algorithm efficiently?

Binary Heap
Union-Find
Stack
Queue

Correct Answer : Union-Find

Ques 202 Which data structure is crucial for efficiently implementing Kruskal's algorithm?

Binary Heap
Union-Find
Stack
Queue

Correct Answer : Union-Find

Ques 203 What does edge relaxation mean in Bellman-Ford algorithm?

Adjusting edge weights
Updating distances for adjacent nodes
Finding cycles in a graph
Adding new edges

Correct Answer : Updating distances for adjacent nodes

Ques 204 How many times are the edges relaxed in Bellman-Ford algorithm in a graph with V vertices?

V
V-1
V+1
E

Correct Answer : V-1

Ques 205 In a graph with 'V' vertices, how many times are the edges relaxed in the Bellman-Ford algorithm?

V
V-1
V+1
E

Correct Answer : V-1

Ques 206 In the Bellman-Ford algorithm, how many times are the edges relaxed for a graph with V vertices?

V
V-1
V+1
E

Correct Answer : V-1

Ques 207 What is the main advantage of Floyd-Warshall algorithm over Dijkstra's algorithm?

Works for all pair shortest paths
Faster in all cases
Simpler to implement
Handles negative weights

Correct Answer : Works for all pair shortest paths

Ques 208 What is the main advantage of the Floyd-Warshall algorithm over Dijkstra's algorithm?

Faster execution
Works for negative weight cycles
Simpler implementation
Works on undirected graphs

Correct Answer : Works for negative weight cycles

Ques 209 What is the primary advantage of Floyd-Warshall Algorithm over Dijkstra's Algorithm?

Faster execution
Works for negative weight cycles
Simpler implementation
Works on undirected graphs

Correct Answer : Works for negative weight cycles

Ques 210 What is the key advantage of Bellman-Ford over Dijkstra's algorithm?

Simpler implementation
Works with negative weights
Faster for dense graphs
Requires fewer iterations

Correct Answer : Works with negative weights

Ques 211 What is the main advantage of the Bellman-Ford algorithm over Dijkstra's algorithm?

Simpler implementation
Works with negative weights
Faster for dense graphs
Requires fewer iterations

Correct Answer : Works with negative weights

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :

Correct Answer :