

DSA Patterns

Viewed better in dark mode.

No.	Pattern	Description	Key Idea	Example
1	Sliding Window	Used for problems involving subarrays or substrings.	Use a sliding window to optimize time complexity from $O(n^2)$ to $O(n)$.	Longest Substring Without Repeating Characters Maximum Sum Subarray of Size K
2	Two Pointers	Used for problems involving sorted arrays, linked lists, or string manipulation.	Use two pointers moving towards/away from each other.	Two Sum (Sorted Array) Trapping Rainwater
3	Fast and Slow Pointers	Used for cyclic problems like finding loops in linked lists.	Use two pointers moving at different speeds.	Linked List Cycle Detection Middle of a Linked List
4	Merge Intervals	Used for problems involving overlapping intervals.	Sort intervals and merge based on conditions.	Merge Intervals Insert Interval
5	Cyclic Sort	Used for problems involving numbers in a given range (1 to n).	Place each number at its correct index.	Missing Number Find All Duplicate Numbers
6	Subsets	Used for problems requiring all combinations/subsets of elements.	Use BFS, recursion, or bitmasking.	Generate All Subsets Generate Permutations
7	Binary Search	Used for searching in sorted arrays or answer-based	Divide and conquer; halve the search space.	Search in Rotated Sorted Array

		problems.		Find Peak Element
8	Backtracking	Used for constraint-based problems like permutations and combinations.	Try all possibilities and backtrack upon failure.	N-Queens Word Search
9	Breadth-First Search (BFS)	Used for shortest path or level-order traversal problems.	Explore all neighbors at the current level before moving to the next level.	Binary Tree Level Order Traversal Word Ladder
10	Depth-First Search (DFS)	Used for pathfinding, tree/graph traversal, and connected components.	Recursively or iteratively explore each path fully.	All Paths in a Graph Number of Islands
11	Topological Sort	Used for problems with dependencies in a Directed Acyclic Graph (DAG).	Use BFS or DFS to order tasks based on prerequisites.	Course Schedule Alien Dictionary
12	Union-Find (Disjoint Set)	Used for connectivity in graphs.	Use union and find operations to manage connected components.	Number of Connected Components Redundant Connection
13	Greedy	Used for optimization problems (minimizing/maximizing something).	Make the locally optimal choice at each step.	Interval Scheduling Huffman Encoding
14	Dynamic Programming (DP)	Used for optimization and decision-based problems.	Break the problem into overlapping subproblems.	Longest Increasing Subsequence 0/1 Knapsack
15	Bit Manipulation	Used for	Use bitwise	Single Number

		binary-related problems like subsets and XOR.	operators to solve efficiently.	Power of Two
16	Matrix Traversal	Used for problems involving grid traversal.	Use BFS, DFS, or dynamic programming.	Unique Paths Rotting Oranges
17	Heap / Priority Queue	Used for problems requiring frequent max/min operations.	Use heaps for efficient insertion and extraction.	Kth Largest Element Merge K Sorted Lists
18	Divide and Conquer	Used for problems involving partitioning.	Divide the problem into smaller subproblem	Merge Sort Median of Two Sorted Arrays
19	Prefix Sum	Used for problems involving range sums.	Precompute cumulative sums to optimize queries.	Subarray Sum Equals K Range Sum Query (Immutable)

20	Sliding Window	Used for problems involving subarrays or substrings.	Use a sliding window to optimize time complexity from $O(n^2)$ to $O(n)$.	Longest Substring Without Repeating Characters Maximum Sum Subarray of Size K
21	Kadane's Algorithm	Used for maximum subarray problems.	Maintain a running sum and update the max sum.	Maximum Subarray Sum Maximum Circular Subarray
22	Trie (Prefix Tree)	Used for word-related problems.	Use a tree structure for fast prefix lookups.	Implement Trie Word Search II
23	Segment Trees	Used for range query problems.	Build a tree structure for efficient range	Range Sum Query Range Minimum

			queries and updates.	Query
24	Graph Traversal	Used for graph-related problems like shortest paths or connected components.	Use DFS, BFS, or Dijkstra's algorithm.	Shortest Path in Weighted Graph Minimum Spanning Tree
25	Flood Fill	Used for grid-based coloring or connected regions.	Use DFS or BFS to visit all connected components	Flood Fill Number of Enclaves
26	Monotonic Stack	Used for problems involving nearest larger/smaller elements.	Use a stack to maintain a monotonic sequence.	Next Greater Element Largest Rectangle in Histogram
27	String Matching (KMP, Rabin-Karp)	Used for finding substrings in a string.	Use efficient string matching algorithms.	Substring Search Shortest Palindrome
28	Binary Indexed Tree (Fenwick Tree)	Used for dynamic range sum/updates.	Use a tree structure to efficiently compute prefix sums.	Range Sum Query Count of Smaller Numbers After Self
29	Reservoir Sampling	Used for random sampling.	Keep track of k items randomly while iterating.	Random Node in a Linked List Random Sampling
30	LRU Cache	Used for caching problems.	Use a hashmap and doubly linked list.	LRU Cache Implementation LFU Cache
31	Fibonacci Sequence	Used for DP problems.	Compute Fibonacci numbers iteratively or using matrix exponentiation.	Climbing Stairs House robbers

32	Game Theory	Used for competitive game problems.	Use minimax or DP to determine the winner.	Nim Game Predict the Winner
33	Mathematical Problems	Used for number theory and combinatorics.	Use properties of numbers for optimization.	GCD and LCM Combination Sum