1. Two Sum (Easy) – Finds two numbers in an array that add up to a target sum. Tests basic array traversal and use of a hash map for quick lookups

fatmaerturk.github.io
.

2. Valid Anagram (Easy) – Checks if two strings are anagrams of each other. Tests sorting or frequency counting of characters (fundamental string manipulation).

3. Valid Parentheses (Easy) – Uses a stack to verify matching brackets in a string. Tests understanding of stack data structure for a classic bracket-validation problem (a staple in coding interviews

blog.unwiredlearning.com
).

4. Merge Two Sorted Lists (Easy) – Merges two sorted linked lists into one sorted list. Tests pointer manipulation in linked lists and handling edge cases (commonly asked in interviews like Amazon

m.youtube.com
).

5. Best Time to Buy and Sell Stock (Easy) – Given daily stock prices, find the max profit from one buy-sell transaction. Tests greedy one-pass array scanning (tracking min price and max profit)

leetcode.com
.

6. Climbing Stairs (Easy) – Counts ways to climb n stairs taking 1 or 2 steps at a time. Tests simple dynamic programming (Fibonacci sequence) for combinatorial reasoning

leetcode.com
.

7. Maximum Subarray (Easy) – Finds the contiguous subarray with the largest sum. Tests dynamic programming/greedy approach (Kadane's algorithm) to accumulate a running maximum

leetcode.com
.

8. Linked List Cycle (Easy) – Detects if a linked list has a cycle using two pointers (Floyd's Tortoise and Hare algorithm). Tests linked list traversal logic and pointer handling

hellointerview.com
.

9. Majority Element (Easy) – Finds the element that appears more than ⌊n/2⌋ times in an array. Tests counting techniques or the Boyer–Moore voting algorithm (known Google interview trick

youtube.com
).

10. Maximum Depth of Binary Tree (Easy) – Computes the height of a binary tree. Tests recursion or BFS/DFS traversal on trees (fundamental tree operation often asked to verify basic tree knowledge).

11. Second Highest Salary (SQL – Easy) – Retrieves the 2nd highest distinct salary from an Employee table. Tests SQL querying with aggregation or subqueries for ranking results (a common SQL interview task

youtube.com
).

12. Combine Two Tables (SQL – Easy) – Performs a simple `JOIN` between two tables (e.g. Person and Address tables). Tests basic SQL JOIN knowledge to combine data from multiple tables in a query.

13. Employees Earning More Than Their Manager (SQL – Medium) – Finds employees who have a salary greater than their manager's salary. Tests self-JOIN on the Employee table to compare an employee to their manager

blog.stackademic.com
.

14. Longest Substring Without Repeating Characters (Medium) – Finds the length of the longest substring with all unique characters. Tests the sliding window and two-pointer technique for string processing (a "standard interview question"

stackoverflow.com
).

15. Product of Array Except Self (Medium) – Returns an array where each element is the product of all other elements in the input array. Tests array manipulation using prefix and suffix products (appears frequently in coding interviews

medium.com
).

16. Group Anagrams (Medium) – Groups words that are anagrams of each other. Tests hashing strategies (sorting characters or counting letter frequencies as keys) to categorize strings (a common interview question

medium.com
).

17. Merge Intervals (Medium) – Merges overlapping intervals in a list of intervals. Tests sorting of intervals and interval merging logic – an algorithm question often encountered in job interviews

finalroundai.com
.

18. Kth Largest Element in an Array (Medium) – Finds the k-th largest number in an unsorted array. Tests use of a min-heap (to keep track of top k elements) or Quickselect partitioning for selection problems (commonly asked for selection algorithms).

19. Top K Frequent Elements (Medium) – Identifies the k most frequent elements in an array. Tests hash map frequency counting and using a heap or bucket sort to retrieve the top frequencies (seen in data science interviews requiring heap + hashtable use

tryexponent.com
).

20. 3Sum (Medium) – Finds all unique triplets in an array that sum up to zero. Tests two-pointer technique on a sorted array to handle combinations and avoid duplicates (a popular tech interview problem

fizzbuzzed.com
).

21. Coin Change (Medium) – Given coin denominations and an amount, determines the fewest coins needed to make that amount (or that it's not possible). Tests dynamic programming for minimizing coin usage – known as one of the more difficult interview DP questions

stratascratch.com
.

22. Add Two Numbers (Medium) – Adds two numbers represented as linked lists (digits stored in reverse order). Tests linked list traversal and handling carry-over in arithmetic (frequently asked by companies like Amazon, Microsoft, Google

fatmaerturk.github.io

23. Search in Rotated Sorted Array (Medium) – Searches for a target value in a sorted array that has been rotated (pivoted). Tests modified binary search logic to handle the rotated order – a common coding challenge to assess binary search skills

medium.com
.

24. LRU Cache (Medium) – Designs a Least Recently Used cache with O(1) get and put operations. Tests knowledge of data structure design (typically implemented with a hash map + doubly-linked list) and is a known interview challenge

youtube.com
.

25. Shuffle an Array (Medium) – Implements an array shuffle (Fisher–Yates algorithm) for randomizing array order. Tests understanding of unbiased random shuffling and use of random number generation (asked in interviews, e.g. as a Google coding question

youtube.com
).

26. Number of Islands (Medium) – Counts the number of “islands” (connected regions of 1s) in a 2D grid of 0s and 1s. Tests graph traversal (DFS/BFS) on a grid to find connected components – a common grid-based interview problem

medium.com
.

27. Word Ladder (Hard) – Given a start word and end word, finds the shortest transformation sequence by changing one letter at a time (each intermediate word must be in a dictionary). Tests breadth-first search on a word graph for shortest paths (a classic BFS interview question

fizzbuzzed.com
).

28. Median of Two Sorted Arrays (Hard) – Finds the median of two sorted arrays combined, in logarithmic time. Tests binary search-based divide-and-conquer logic to partition arrays around the median – famously asked at companies like Google and Apple

youtube.com
.

29. Find Median from Data Stream (Hard) – Continuously maintains the median of a growing number stream. Tests use of two heaps (a max-heap for lower half and min-heap for upper half of numbers) to efficiently retrieve the median on the fly

youtube.com
.

30. Merge K Sorted Lists (Hard) – Merges k sorted linked lists into one sorted linked list. Tests efficient merging using a min-heap (priority queue) or divide-and-conquer merging, to handle multiple sorted data streams (commonly asked in interviews, e.g. Amazon

ambitionbox.com
)