

Mastering the Coding Interview (Big Tech) - Course Outline

zerotomastery.io

Arrays

Question #1 Google Interview Question Two Sum (Easy)

- [Two Sum Leetcode Question](#)
- [Brute Force Solution Code Repl](#)
- [Optimal Solution Code Repl](#)

Question #2 Container With Most Water (Medium)

- [Container With Most Water Leetcode Question](#)
- [Brute Force Solution Code Repl](#)
- [Optimal Solution Code Repl](#)

Question #3 Trapping Rainwater (Hard)

- [Trapping Rainwater Leetcode Question](#)
- [Brute Force Solution Code Repl](#)
- [Optimal Solution Code Repl](#)

Strings

Question #4 Backspace String Compare (Easy)

- [Typed Out Strings Leetcode Question](#)
- [Brute Force Solution Code Repl](#)
- [Optimal Solution Code Repl](#)

Question #5 Longest Substring Without Repeating Characters (Medium)

- [Longest Substring Without Repeating Characters Leetcode Question](#)
- [Brute Force Solution Code Repl](#)
- [Optimal Solution Code Repl](#)

Palindromes

Question #6a Valid Palindrome(Easy)

- [Valid Palindrome Leetcode Question](#)
- [Solution #1 - Two Pointers From The Center Solution](#)
- [Solution #2 - Two Pointers From The Outside Solution](#)
- [Solution #3 - Compare Against Reverse Solution](#)

Question #6b Almost Palindrome (Easy)

- [Almost Palindrome Leetcode Question](#)
- [Final Code Solution Repl](#)

Linked Lists

Basic Algorithm - Reverse Linked List

- [Code Solution Repl](#)

Question #7 M, N Reversals (Medium)

- [M, N Reversals Leetcode Question](#)
- [Code Solution Repl](#)

Doubly Linked Lists

Question #8 Merge Multi-Level Doubly Linked List (Medium)

- [Merge Multi-Level Doubly Linked List Leetcode Question](#)
- [Code Solution Repl](#)

Question #9 Cycle Detection (Medium)

- [Cycle Detection Leetcode Question](#)
- [Floyd's Tortoise and Hare Algorithm Code Repl](#)
- [Code solution with Set Object Code Repl](#)

Stacks

Question #10 Valid Parentheses (Easy)

- [Valid Parentheses Leetcode Question](#)
- [Code Solution Repl](#)

Question #11 Minimum Brackets To Remove To Make Valid (Medium)

- [Minimum Brackets To Remove To Make Valid Leetcode Question](#)
- [Code Solution Repl](#)

Queues

Question #12 Implement Queue With Stacks (Easy)

- [Implement Queue With Stacks Leetcode Question](#)
- [Code Solution Repl](#)

Recursion

Sorting

Question #13 Kth Largest Element (Medium)

- [Kth Largest Element Leetcode Question](#)
- [Quicksort Solution Repl](#)
- [Hoare's QuickSelect Solution Repl](#)

Binary Search

Question #14 Start And End Of Target (Medium)

- [Start And End Of Target Leetcode Question](#)
- [Code Solution Repl](#)

Binary Trees

Question #15 Maximum Depth Of Binary Tree (Easy)

- [Maximum Depth Of Binary Tree Leetcode Question](#)
- [Code Solution Repl](#)

Question #16 Level Order Of Binary Tree (Medium)

- [Level Order Of Binary Tree Leetcode Question](#)
- [Code Solution Repl](#)

Question #17 Right Side View of Tree (Medium)

- [Right Side View of Tree Leetcode Question](#)
- [BFS Solution Code Repl](#)
- [DFS Solution Code Repl](#)

Full And Complete Binary Trees

Question #18 Number Of Nodes In Complete Tree (Medium)

- [Number Of Nodes In Complete Tree Leetcode Question](#)
- [Code Solution Repl](#)

Binary Search Trees

Question #19 Validate Binary Search Tree (Medium)

- [Validate Binary Search Tree Leetcode Question](#)
- [Code Solution Repl](#)

Heaps & Priority Queues

Max Heap/ Priority Queue

- [Max Heap/ Priority Queue Code Repl](#)

2D-Arrays

Traversals

DFS in 2D-Arrays

- [DFS Code Repl](#)

BFS in 2D-Arrays

- [BFS Code Repl](#)

Question #20 Number Of Islands (Medium)

- [Number Of Islands Leetcode Question](#)
- [BFS Solution Code Repl](#)
- [DFS Solution Code Repl](#)

Question #21 Rotting Oranges (Medium)

- [Rotting Oranges Leetcode Question](#)
- [Code Solution Repl](#)

Question #22 Walls And Gates (Medium)

- [Walls And Gates Leetcode Question \(Subscription Required\)](#)
- [Code Solution Repl](#)

Graphs

BFS Graph Traversal

- [Adjacency List BFS Code Repl](#)
- [Adjacency Matrix BFS Code Repl](#)

DFS Graph Traversals

- [Adjacency List DFS Code Repl](#)
- [Adjacency Matrix DFS Code Repl](#)

Question #23 Time Needed To Inform All Employees (Medium)

- [Time Needed To Inform All Employees Leetcode Question](#)
- [Code Solution Repl](#)

Question #24 Course Scheduler (Medium)

- [Course Scheduler Leetcode Question](#)
- [Naive BFS Solution Code Repl](#)
- [Topological Sort Solution With Adjacency List Code Repl](#)
- [Optimal Topological Sort Solution \(No Adjacency List\) Code Repl](#)
- [Optimal Topological Sort Solution Modified To Return Order Code Repl](#)

Question #25 Network Time Delay (Medium)

- [Network Time Delay Leetcode Question](#)
- [Dijkstra's Algorithm Solution Code Repl](#)
- [Bellman-Ford Solution Code Repl](#)

Dynamic Programming

Question #26 Minimum Cost Of Climbing Stairs (Easy)

- [Minimum Cost Of Climbing Stairs Leetcode Question](#)
- [Top Down Recursive Solution Code Repl](#)
- [Top Down With Memoization Solution Code Repl](#)
- [Bottom Up Iterative Solution Code Repl](#)
- [Bottom Up Optimized Solution Code Repl](#)

Question #27 Knight Probability In Chessboard (Medium)

- [Knight Probability In Chessboard Leetcode Question](#)
- [Top Down Recursive Solution Code Repl](#)
- [Top Down With Memoization Solution Code Repl](#)
- [Bottom Up Iterative Solution Code Repl](#)
- [Bottom Up Optimized Solution Code Repl](#)

Back Tracking

Question #28 Sudoku Solver (Hard)

- [Sudoku Solver Leetcode Question](#)
- [Backtracking Solution Code Repl](#)

Additional Backtracking Problems:

Palindrome Partitioning (Medium)

- [Palindrome Partitioning Leetcode Question](#)
- [Backtracking Solution Code Repl](#)

N-Queens (Hard)

- [N-Queens Leetcode Question](#)
- [Backtracking Solution Code Repl](#)

Interface Design

Question #29 Monarchy (Medium)

- [Monarchy Implementation Code Repl](#)

Tries

Question #30 Implement Prefix Trie (Medium)

- [Implement Prefix Trie Leetcode Question](#)
- [Trie Class Implementation Code Repl](#)