

Mastering Coding Test

50 Problems with
Solutions

'Goke
Akintoye

Mastering Coding Test

50 Problems with
Solutions

‘Goke
Akintoye

All rights reserved. No part of this book may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, or by any information storage and retrieval system, without permission in writing from the publisher.

Copyright © 2024 by Adegoke Akintoye

First edition. March 17, 2024.

Juri Books: call.juri@outlook.com/+2349012885870

Disclaimer

While every precaution has been taken in the preparation of this book, the publisher assumes no responsibility for errors or omissions, or for damages resulting from the use of the information contained herein.

Other Books By The Author:

1. [Mastering JavaScript Array Methods: A Beginner's Guide to Simplifying Array Manipulation](#)
2. [Mastering JavaScript String Methods: A Beginner's Guide to Simplifying String Manipulation](#)
3. [Mastering Coding Test: 50 Problems with Solutions](#)
4. [Mastering Design Patterns in TypeScript: An Approachable Guide](#)
5. [Object-Oriented Programming In TypeScript: A Beginner's Guide](#)
6. [Mastering TypeScript: A Beginner's Guide](#)
7. [JavaScript: The Ultimate Guide to Interview Questions](#)
8. [Integrating HTMX with Laravel: An Approachable Guide](#)
9. [Object-Oriented Programming in PHP: An Approachable Guide](#)
10. [Functional Programming in TypeScript: An Approachable Guide](#)
11. [Laravel Guide](#)
12. [Mastering API Development with Laravel](#)
13. [HTMX Guide](#)
14. [Lumen Illuminated](#)
15. [Easy, Fast, and Practical PWA](#)

Table of Contents

Disclaimer.....	4
While every precaution has been taken in the preparation of this book, the publisher assumes no responsibility for errors or omissions, or for damages resulting from the use of the information contained herein.....	4
Other Books By The Author:.....	5
Content.....	6
Introduction.....	12
Challenge 1 : Two Sum.....	13
Problem Description:.....	13
Input:.....	13
Output:.....	13
Solution (JavaScript):.....	13
Example:.....	14
Complexity Analysis:.....	14
Challenge 2. Container With Most Water.....	14
Problem Description:.....	15
Input:.....	15
Output:.....	15
Solution:.....	15
Complexity Analysis:.....	16
Challenge 3. Longest Substring Without Repeating Characters.....	16
Problem Description:.....	16
Input 1:.....	16
Output 1:.....	16
Input 2:.....	17
Output 2:.....	17
Solution:.....	17
Explanation:.....	17
Complexity Analysis:.....	18
Challenge 4. Reverse Integer.....	18
Problem Description:.....	18
Input/Output Example(s):.....	18
Solution:.....	18
Complexity Analysis:.....	19
Challenge 5. Valid Palindrome.....	19
Problem Description:.....	20
Input/Output Example(s):.....	20
Solution:.....	20
Complexity Analysis:.....	21
Challenge 6 : Merge Two Sorted Lists.....	21
Problem Description:.....	21
Input/Output Example(s):.....	21
Solution :.....	21
Complexity Analysis:.....	24
Challenge 7 : Add Two Numbers.....	24
Problem Description:.....	24
Input/Output Example(s):.....	24

Solution:.....	24
Complexity Analysis:.....	26
Challenge 8 : The Greatest Common Divisor of Strings.....	26
Problem Description:.....	26
Input/Output Example(s):.....	26
Solution:.....	26
Complexity Analysis:.....	27
Challenge 9 : Merge Strings Alternately.....	28
Problem Description:.....	28
Input/Output Example(s):.....	28
Solution:.....	29
Complexity Analysis:.....	30
Challenge 10 : Kids With the Greatest Number of Candies.....	30
Problem Description:.....	30
Input/Output Example(s):.....	30
Solution:.....	31
Complexity Analysis:.....	32
Challenge 11 : Can Place Flowers.....	32
Problem Description:.....	33
Input/Output Example(s):.....	33
Solution:.....	33
Complexity Analysis:.....	34
Challenge 12 : Reverse Vowels of a String.....	35
Problem Description:.....	35
Input/Output Example(s):.....	35
Solution:.....	35
Complexity Analysis:.....	36
Challenge 13 : Product of Array Except Self.....	37
Problem Description:.....	37
Input/Output Example(s):.....	37
Solution:.....	37
Complexity Analysis:.....	38
Challenge 14 : Increasing Triplet Subsequence.....	39
Problem Description:.....	39
Input/Output Example(s):.....	39
Solution:.....	40
Complexity Analysis:.....	41
Challenge 15 : Remove Nth Node From End of List.....	41
Problem Description:.....	41
Input/Output Example(s):.....	41
Solution:.....	41
Complexity Analysis:.....	43
Challenge 16 : Reverse Linked List.....	43
Problem Description:.....	43
Input/Output Example(s):.....	43
Solution:.....	44
Complexity Analysis:.....	45
Challenge 17 : Detect Cycle in a Linked List.....	45
Problem Description:.....	45

Input/Output Example(s):.....	45
Solution:.....	46
Complexity Analysis:.....	47
Challenge 18 : Binary Tree Inorder Traversal.....	47
Problem Description:.....	47
Input/Output Example(s):.....	47
Solution:.....	48
Complexity Analysis:.....	49
Challenge 19 : Maximum Depth of Binary Tree.....	49
Problem Description:.....	49
Input/Output Example(s):.....	49
Solution:.....	50
Complexity Analysis:.....	51
Challenge 20 : Merge Two Binary Trees.....	51
Problem Description:.....	51
Input/Output Example(s):.....	51
Solution:.....	52
Complexity Analysis:.....	53
Challenge 21 : Course Schedule.....	53
Problem Description:.....	53
Input/Output Example(s):.....	53
Solution:.....	53
Complexity Analysis:.....	55
Challenge 22 : Number of Islands.....	55
Problem Description:.....	55
Input/Output Example(s):.....	55
Solution:.....	56
Complexity Analysis:.....	57
Challenge 23 : Climbing Stairs.....	57
Problem Description:.....	58
Input/Output Example(s):.....	58
Solution:.....	58
Complexity Analysis:.....	59
Challenge 24 : Longest Common Subsequence.....	59
Problem Description:.....	59
Input/Output Example(s):.....	60
Solution:.....	60
Complexity Analysis:.....	61
Challenge 25 : Maximum Subarray.....	62
Problem Description:.....	62
Input/Output Example(s):.....	62
Solution:.....	62
Complexity Analysis:.....	63
Challenge 26 : Coin Change.....	63
Problem Description:.....	63
Input/Output Example(s):.....	64
Solution:.....	64
Complexity Analysis:.....	65
Challenge 27 : Word Break.....	65

Problem Description:.....	66
Input/Output Example(s):.....	66
Solution:.....	66
Complexity Analysis:.....	67
Challenge 28 : Permutations.....	67
Problem Description:.....	67
Input/Output Example(s):.....	67
Solution:.....	68
Complexity Analysis:.....	69
Challenge 29 : Combination Sum.....	70
Problem Description:.....	70
Input/Output Example(s):.....	70
Solution:.....	70
Complexity Analysis:.....	72
Challenge 30 : N-Queens.....	72
Problem Description:.....	72
Input/Output Example(s):.....	72
Solution:.....	72
Complexity Analysis:.....	75
Challenge 31 : Generate Parentheses.....	75
Problem Description:.....	75
Input/Output Example(s):.....	75
Solution:.....	75
Complexity Analysis:.....	77
Challenge 32 : Word Search.....	77
Problem Description:.....	77
Input/Output Example(s):.....	77
Solution:.....	78
Complexity Analysis:.....	79
Challenge 33 : Merge Intervals.....	79
Problem Description:.....	79
Input/Output Example(s):.....	79
Solution:.....	80
Complexity Analysis:.....	81
Challenge 34 : Kth Largest Element in an Array.....	81
Problem Description:.....	81
Input/Output Example(s):.....	82
Solution:.....	82
Complexity Analysis:.....	83
Challenge 35 : Search in Rotated Sorted Array.....	84
Problem Description:.....	84
Input/Output Example(s):.....	84
Solution:.....	84
Complexity Analysis:.....	86
Challenge 36 : Find First and Last Position of Element in Sorted Array.....	86
Problem Description:.....	86
Input/Output Example(s):.....	86
Solution:.....	87
Complexity Analysis:.....	89

Challenge 37 : Search a 2D Matrix.....	89
Problem Description:.....	89
Input/Output Example(s):.....	89
Solution:.....	89
Complexity Analysis:.....	91
Challenge 38 : Implement Stack using Queues.....	91
Problem Description:.....	91
Input/Output Example(s):.....	92
Solution:.....	92
Complexity Analysis:.....	94
Challenge 39 : Min Stack.....	94
Problem Description:.....	94
Input/Output Example(s):.....	94
Solution:.....	95
Complexity Analysis:.....	96
Challenge 40 : Design Twitter.....	97
Problem Description:.....	97
Input/Output Example(s):.....	97
Solution:.....	97
Complexity Analysis:.....	100
Challenge 41 : Single Number.....	100
Problem Description:.....	100
Input/Output Example(s):.....	100
Solution:.....	100
Complexity Analysis:.....	101
Challenge 42 : Counting Bits.....	101
Problem Description:.....	101
Input/Output Example(s):.....	101
Solution:.....	102
Complexity Analysis:.....	103
Challenge 43 : Majority Element.....	103
Problem Description:.....	103
Input/Output Example(s):.....	103
Solution:.....	104
Complexity Analysis:.....	105
Challenge 44 : Power of Two.....	105
Problem Description:.....	105
Input/Output Example(s):.....	105
Solution:.....	105
Complexity Analysis:.....	106
Challenge 45 : Reverse Bits.....	107
Problem Description:.....	107
Input/Output Example(s):.....	107
Solution:.....	107
Complexity Analysis:.....	108
Challenge 46 : URL Shortener.....	108
Problem Description:.....	109
Input/Output Example(s):.....	109
Solution:.....	109

Complexity Analysis:.....	110
Challenge 47 : Design a Parking Lot.....	111
Problem Description:.....	111
Input/Output Example(s):.....	111
Solution:.....	111
Complexity Analysis:.....	114
Challenge 48 : Map Reduce.....	114
Problem Description:.....	114
Input/Output Example(s):.....	114
Solution:.....	115
Complexity Analysis:.....	116
Challenge 49 : Cache design for high traffic.....	116
Problem Description:.....	117
Input/Output Example(s):.....	117
Complexity Analysis:.....	119
Challenge 50 :.....	119
Problem Description:.....	119
Input/Output Example(s):.....	119
Solution:.....	120
Complexity Analysis:.....	121

Introduction

Welcome to “Mastering Coding Test: 50 Problems with Solutions.”

This book aims to provide a curated list of 50 of the most popular coding/interview test/problems along with detailed explanations and solutions to help you master these concepts and improve your problem-solving skills.

This book assumes basic understanding of JavaScript.

Your feedback will be highly appreciated. You can reach me here:

call.juri@outlook.com

Challenge 1 : Two Sum

Problem Description:

The Two Sum problem is a classic algorithmic problem that can be solved using multiple approaches. The problem statement is as follows: Given an array of integers `nums` and an integer `target`, return indices of two numbers such that they add up to `target`. You may assume that each input would have exactly one solution, and you may not use the same element twice.

Input:

- `nums`: Array of integers - The array of integers where we need to find two elements that sum up to the target.
- `target`: Integer - The target sum we are looking for.

Output:

- `result`: Array of integers (The indices of the two numbers that add up to the target).

Solution (JavaScript):

Here is a simple solution using a hash map to store the indices of elements we have seen so far:

```
1  function twoSum(nums, target) {
2      let hashmap = new Map();
3
4      for (let i = 0; i < nums.length; i++) {
5          const complement = target - nums[i];
6          if (hashmap.has(complement)) {
7              return [hashmap.get(complement), i];
8          }
9          hashmap.set(nums[i], i);
10     }
```

```
11 }
```

Example:

Let's consider an example input and see how the function works:

```
1  const nums = [2, 7, 11, 15];  
2  const target = 9;  
3  
4  console.log(twoSum(nums, target)); // Output: [0, 1]
```

In this example, the function `twoSum` is called with `nums` array and target sum of 9. The function returns the indices `[0, 1]` which correspond to the numbers 2 and 7 in the `nums` array that add up to the target sum of 9.

Complexity Analysis:

- Time Complexity: $O(n)$ - We traverse the array of size n only once.
- Space Complexity: $O(n)$ - The space used by the hashmap is at most n (in the worst case when all elements are unique).

Challenge 2. Container With Most Water

To solve the Container With Most Water problem, we are given an array of non-negative integers representing the heights of walls, we need to find the maximum area of water that can be trapped between two walls by selecting two walls whose height and distance maximize the area.

Here is a detailed description and solution in JavaScript:

Problem Description:

Given an array of non-negative integers representing the heights of walls, we need to find the maximum area of water that can be trapped between two walls.

Input:

- An array of non-negative integers representing the heights of walls.

Output:

- Maximum area of water that can be trapped.

Solution:

```
1  const maxArea = (height) => {
2      let maxArea = 0;
3      let left = 0;
4      let right = height.length - 1;
5
6      while (left < right) {
7          const minHeight = Math.min(height[left],
height[right]);
8          maxArea = Math.max(maxArea, (right - left) *
minHeight);
9
10         if (height[left] < height[right]) {
11             left++;
12         } else {
13             right--;
14         }
15     }
```

```

16
17     return maxArea;
18 };
19
20 // Example input and output
21 const height1 = [1, 8, 6, 2, 5, 4, 8, 3, 7];
22 console.log(maxArea(height1)); // Output: 49
23
24 const height2 = [4, 3, 2, 1, 4];
25 console.log(maxArea(height2)); // Output: 16

```

In this solution, we use a two-pointer approach where we start with two pointers at the leftmost and rightmost walls of the array. We calculate the area between the two walls by taking the minimum height of the two walls and multiplying it by the distance between them. We update the maximum area found so far and move the pointer with the smaller height towards the center, as a larger height wall might potentially give us a bigger area.

Complexity Analysis:

- The **time complexity** of this solution is **$O(n)$** as we only iterate through the array once.
- The **space complexity** is **$O(1)$** as we only use a constant amount of extra space.

This approach efficiently solves the Container With Most Water problem by finding the maximum area that can be trapped between the two walls given the array of wall heights.