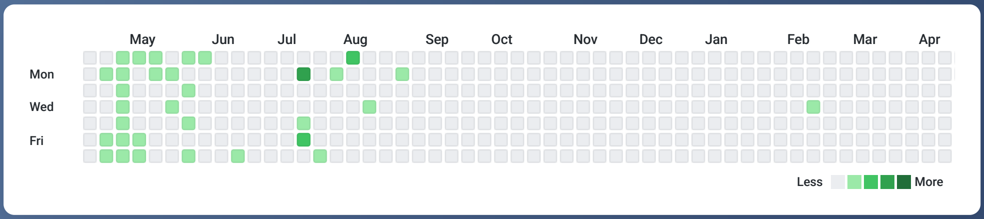


7
Watch

162
Star

83
Fork

3
Issue

overview activity issues

A collection of solutions for all problem statements on the AlgoExpert Coding Interview platform.

7 PYTHON

das-jishu master

pushedAt 1 month ago

das-jishu

algorithms data-structures algoexpert problem-solving algorithms-and-data-structures

das-jishu/algoexpert-data-structures-algorithms

⭐ algo-expert

algoExpert

License MIT | repo size 9.84 MB | Status Active | lang python | last updated 28-08-2021

Description: This is a collection of all AlgoExpert Coding Interview questions that are currently available on the platform. There are solutions for each problem statement including time and space complexity. Since AlgoExpert is a paid platform, everyone doesn't have access to it or can't afford to. I hope this helps everyone to access the content and improve their problem solving skills.

Problem List + Solutions • How to • Contribute • Extras

Solutions: The solutions are provided in Python.

Go to Top

⚡ LIST OF PROBLEMS

⭐ : Coding Interview Problems

Difficulty chart:

🟩 : Easy
🟦 : Medium
🟥 : Hard
⬛ : Very Hard

	Problem Statement	Difficulty	Solution
⭐	Two Number Sum	🟩	two-number-sum.py
⭐	Validate Subsequence	🟩	validate-subsequence.py
⭐	Tournament Winner	🟩	tournament-winner.py
⭐	Non-Constructible Change	🟩	non-constructible-change.py
⭐	Find Closest Value in BST	🟩	closest-in-BST.py
⭐	Branch Sums	🟩	branch-sums.py
⭐	Node Depths	🟩	node-depths.py
⭐	Depth First Search	🟩	depth-first-search.py
⭐	Minimum Waiting Time	🟩	minimum-waiting-time.py
⭐	Class Photos	🟩	class-photos.py
⭐	Remove Duplicates from Linked List	🟩	remove-duplicates.py
⭐	Nth Fibonacci	🟩	nth-fibonacci.py
⭐	Validate Subsequence	🟩	validate-subsequence.py
⭐	Product Sum	🟩	product-sum.py
⭐	Binary Search	🟩	binary-search.py
⭐	Find Three Largest Numbers	🟩	find-three-largest-numbers.py
⭐	Bubble Sort	🟩	bubble-sort.py
⭐	Insertion Sort	🟩	insertion-sort.py
⭐	Selection Sort	🟩	selection-sort.py
⭐	Palindrome Check	🟩	palindrome-check.py
⭐	Caesar Cipher Encryptor	🟩	caesar-cipher-encryptor.py
⭐	Run Length Encoding	🟩	run-length-encoding.py
⭐	Generate Document	🟩	generate-document.py
⭐	Sorted Square Array	🟩	sorted-square-array.py
⭐	First Non Repeating Character	🟩	first-non-repeating-character.py
⭐	Tandem Bicycle	🟩	tandem-bicycle.py
⭐	Three Number Sum	🟦	three-number-sum.py
⭐	Smallest Difference	🟦	smallest-difference.py
⭐	Move Element to End	🟦	move-element-to-end.py
⭐	Monotonic Array	🟦	monotonic-array.py
⭐	Spiral Traverse	🟦	spiral-traverse.py
⭐	Longest Peak	🟦	longest-peak.py
⭐	Array of Products	🟦	array-of-products.py

⭐	First Duplicate Value	🟦	first-duplicate-value.py
⭐	Merge Overlapping Intervals	🟦	merge-overlapping-intervals.py
⭐	BST Construction	🟦	bst-construction.py
⭐	Validate BST	🟦	validate-bst.py
⭐	BST Traversal	🟦	bst-traversal.py
⭐	Min Height BST	🟦	min-height-bst.py
⭐	Find Kth Largest Value In BST	🟦	find-kth-largest-value-in-bst.py
⭐	Reconstruct BST	🟦	reconstruct-bst.py
⭐	Invert Binary Tree	🟦	invert-binary-tree.py
⭐	Binary Tree Diameter	🟦	binary-tree-diameter.py
⭐	Height Balanced Binary Tree	🟦	height-balanced-binary-tree.py
⭐	Max Subset Sum No Adjacent	🟦	max-subset-sum-no-adjacent.py
⭐	Number of Ways to Make Change	🟦	ways-to-make-change.py
⭐	Min Number of Coins for Change	🟦	min-number-of-coins-for-change.py
⭐	Levenshtein Distance	🟦	levenshtein-distance.py
⭐	Kadane's Algorithm	🟦	kadane's-algorithm.py
⭐	Single Cycle Check	🟦	single-cycle-check.py
⭐	Breadth First Search	🟦	breadth-first-search.py
⭐	Youngest Common Ancestor	🟦	youngest-common-ancestor.py
⭐	Remove Islands	🟦	remove-islands.py
⭐	Cycle In Graph	🟦	cycle-in-graph.py
⭐	Minimum Passes of Matrix	🟦	minimum-passes-of-matrix.py
⭐	Task Assignment	🟦	task-assignment.py
⭐	Valid Starting City	🟦	valid-starting-city.py
⭐	Min Heap Construction	🟦	min-heap-construction.py
⭐	Linked List Construction	🟦	linked-list-construction.py
⭐	Remove Kth Node From End	🟦	remove-kth-node-from-end.py
⭐	Sum of Linked Lists	🟦	sum-of-linked-lists.py
⭐	Permutations	🟦	permutations.py
⭐	Powerset	🟦	powerset.py
⭐	Phone Number Mnemonics	🟦	phone-number-mnemonics.py
⭐	Staircase Traversal	🟦	staircase-traversal.py
⭐	Search in Sorted matrix	🟦	search-in-sorted-matrix.py
⭐	Three Number Sort	🟦	three-number-sort.py
⭐	Min Max Stack construction	🟦	min-max-stack-construction.py
⭐	Balanced Brackets	🟦	balanced-brackets.py
⭐	Sunset Views	🟦	sunset-views.py
⭐	Sort Stack	🟦	sort-stack.py
⭐	Next Greater Element	🟦	next-greater-element.py
⭐	Group Anagrams	🟦	group-anagrams.py
⭐	Valid IP Addresses	🟦	valid-ip-addresses.py
⭐	Reverse Words In String	🟦	reverse-words-in-string.py
⭐	Minimum Characters For Words	🟦	minimum-characters-for-words.py
⭐	Suffix Trie Construction	🟦	suffix-trie-construction.py
⭐	Four Number Sum	🟥	four-number-sum.py
⭐	Subarray Sort	🟥	subarray-sort.py
⭐	Largest Range	🟥	largest-range.py
⭐	Min Rewards	🟥	min-rewards.py
⭐	Zigzag Traverse	🟥	zigzag-traverse.py

⭐	Zigzag Traverse	🟥	zigzag-traverse.py
⭐	Same bsts	🟥	same-bsts.py
⭐	Validate Three Nodes	🟥	validate-three-nodes.py
⭐	Max Path Sum In Binary Tree	🟥	max-path-sum-in-binary-tree.py
⭐	Find Nodes Distance K	🟥	find-nodes-distance-k.py
⭐	Longest Common Subsequence	🟥	longest-common-subsequence.py
⭐	Min Number of Jumps	🟥	min-number-of-jumps.py
⭐	Water Area	🟥	water-area.py
⭐	Knapsack Problem	🟥	knapsack-problem.py
⭐	Disk Stacking	🟥	disk-stacking.py
⭐	Numbers in Pi	🟥	numbers-in-pi.py
⭐	Maximum Sum Submatrix	🟥	maximum-sum-submatrix.py
⭐	Dijkstra Algorithm	🟥	dijkstra-algorithm.py
⭐	Topological Sort	🟥	topological-sort.py
⭐	Boggle Board	🟥	boggle-board.py
⭐	Continuous Median	🟥	continuous-median.py
⭐	Sort K Sorted Array	🟥	sort-k-sorted-array.py
⭐	Laptop Rentals	🟥	laptop-rentals.py
⭐	Find Loop	🟥	find-loop.py
⭐	Reverse Linked List	🟥	reverse-linked-list.py
⭐	Merge Linked Lists	🟥	merge-linked-lists.py
⭐	Shift Linked Lists	🟥	shift-linked-lists.py
⭐	Lowest Common Manager	🟥	lowest-common-manager.py
⭐	Solve Sudoku	🟥	solve-sudoku.py
⭐	Generate Div Tags	🟥	generate-div-tags.py
⭐	Ambiguous Measurements	🟥	ambiguous-measurements.py
⭐	Shifted Binary Search	🟥	shifted-binary-search.py
⭐	Search For Range	🟥	search-for-range.py
⭐	Quickselect	🟥	quickselect.py
⭐	Index Equals Value	🟥	index-equals-value.py
⭐	Quick Sort	🟥	quick-sort.py
⭐	Heap Sort	🟥	heap-sort.py
⭐	Radix Sort	🟥	radix-sort.py
⭐	Shorten Path	🟥	shorten-path.py
⭐	Largest Rectangle Under Skyline	🟥	largest-rectangle-under-skyline.py
⭐	Longest Substring Without Duplication	🟥	longest-substring-without-duplication.py
⭐	Underscorify Substring	🟥	underscorify-substring.py
⭐	Pattern Matcher	🟥	pattern-matcher.py
⭐	Multi String Search	🟥	multi-string-search.py
⭐	Apartment Hunting	⬛	apartment-hunting.py
⭐	Calendar Matching	⬛	calendar-matching.py
⭐	Waterfall Streams	⬛	waterfall-streams.py
⭐	Minimum Area Rectangle	⬛	minimum-area-rectangle.py
⭐	Line Through Points	⬛	line-through-points.py
⭐	Right Smaller Than	⬛	right-smaller-than.py
⭐	Iterative Inorder Traversal	⬛	iterative-inorder-traversal.py
⭐	Flatten Binary Tree	⬛	flatten-binary-tree.py
⭐	Right Sibling Tree	⬛	right-sibling-tree.py
⭐	All Kinds of Node Depths	⬛	all-kinds-of-node-depths.py
⭐	Compare Leaf Traversal	⬛	compare-leaf-traversal.py

⭐	Compare Leaf Traversal	⬛	compare-leaf-traversal.py
⭐	Max Profits With K Transactions	⬛	max-profits-with-k-transactions.py
⭐	Palindrome Partitioning Min Cuts	⬛	palindrome-partitioning-min-cuts.py
⭐	Longest Increasing Subsequence	⬛	longest-increasing-subsequence.py
⭐	Longest String Chain	⬛	longest-string-chain.py
⭐	Square Of Zeroes	⬛	square-of-zeroes.py
⭐	A Star Algorithm	⬛	A-star-algorithm.py
⭐	Detect Arbitrage	⬛	detect-arbitrage.py
⭐	Airport Connections	⬛	airport-connections.py
⭐	Merge Sorted Arrays	⬛	merge-sorted-arrays.py
⭐	LRU Cache	⬛	LRU-cache.py
⭐	Rearrange Linked List	⬛	rearrange-linked-list.py
⭐	Linked List Palindrome	⬛	linked-list-palindrome.py
⭐	Zip Linked List	⬛	zip-linked-list.py
⭐	Node Swap	⬛	node-swap.py
⭐	Number of Binary Tree Topologies	⬛	number-of-binary-tree-topologies.py
⭐	Non Attacking Queens	⬛	non-attacking-queens.py
⭐	Merge Sort	⬛	merge-sort.py
⭐	Count Inversions	⬛	count-inversions.py
⭐	Longest Balanced Substring	⬛	longest-balanced-substring.py

Go to Top

⚡ HOW TO

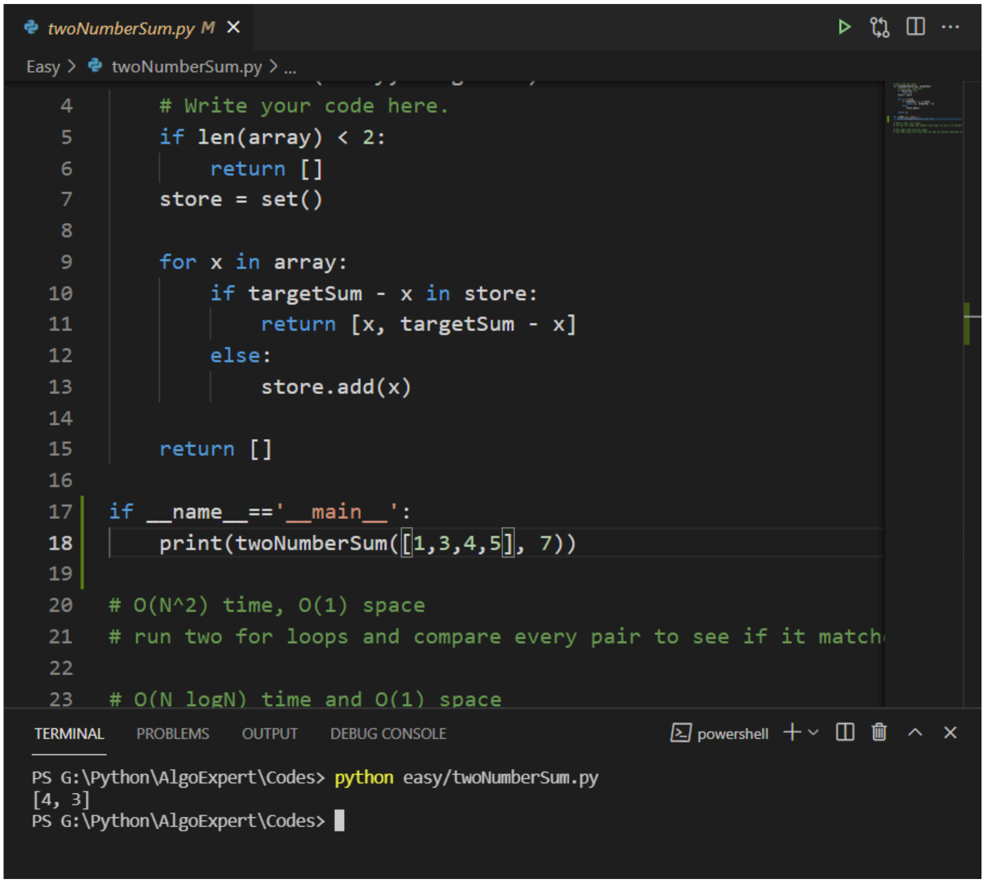
You can visit this page and download the latest Python release version: Install Python
After you complete the download and install, you can run any solution by writing a 'main' section in the file so that the compiler knows the sequence of methods to call when executing the script.

For example, in case of 'Two Number Sum' problem:

```
if __name__=='__main__':
        print(twoNumberSum([1,3,4,5], 7))
```

The code can be executed like this:

```
python easy/twoNumberSum.py
```



```python
    # Write your code here.
    if len(array) < 2:
        return []
    store = set()

    for x in array:
        if targetSum - x in store:
            return [x, targetSum - x]
        else:
            store.add(x)

    return []

if __name__=='__main__':
    print(twoNumberSum([1,3,4,5], 7))

# O(N^2) time, O(1) space
# run two for loops and compare every pair to see if it match

# O(N logN) time and O(1) space
```

```
PS G:\Python\AlgoExpert\Codes> python easy/twoNumberSum.py
[4, 3]
PS G:\Python\AlgoExpert\Codes>
```

Go to Top

⚡ CONTRIBUTE, CONTRIBUTE, CONTRIBUTE!

This is not near to perfect. So please feel free to fork this repo and add any solution in different languages here. You can even add test cases to make this robust. Let's help each other grow! 😀

Go to Top

⚡ EXTRAS

I also have a collection of Leetcode questions that I keep growing. Feel free to visit and have a look. It has more than 200 questions + solutions and also basic concepts grouped by category. 🔽

Leetcode Material and Basics