Question	Link	Technique	Categories	Solved	https://leetcode.com/discuss/general-discussion/460599/blind-75-leetcode-questions
Two Sum	https://leetcode.com/problems/two-sum/submissions/	Dictionary to keep track if the ind	Array	Yes	
Best time to buy and sell stock	https://leetcode.com/problems/best-time-to-buy-and-sell-stock/	use if to find left and right bound i	Array	Yes	
contains duplicate	https://leetcode.com/problems/contains-duplicate/submissions/	use set to eliminate duplicates	Array	Yes	
Product of array except self	https://leetcode.com/problems/product-of-array-except-self/		Array	No	

Reversed Linked List	https://leetcode.com/problems/reverse-linked-list/	set prev to None
		nextv = head.next #2
		head.next = prev #1->Nor
		prev = head #prev = 1
		head = nextv #head = 2
			LinkedList	Yes	

linkedlist cycle	https://leetcode.com/problems/linked-list-cycle/submissions/	slow pointer and fast pointer	LinkedList	Yes	
Merge two sorteed lists	https://leetcode.com/problems/merge-two-sorted-lists/submissions/	1. get a head first by comparing t 2. while loop to compare all the e 3. if one of the list is None, apper 4. check if list1 is none return list2	LinkedList	Yes	
Maximum subarray	https://leetcode.com/problems/maximum-subarray/	https://leetcode.com/problems/ma	Array	Yes	
Maximum Product Subarray	https://leetcode.com/problems/maximum-product-subarray/		Array	No	
Find Minimum in Rotated Sorted A	https://leetcode.com/problems/find-minimum-in-rotated-sorted-array/	1. compare the value of left and r 2. do it like binary search.	Array	Yes	
Search in Rotated Sorted Array	https://leetcode.com/problems/search-in-rotated-sorted-array/	1. compare the value of left and r 2. if left > mid: search mid to high	Array	Yes	
3 Sum	https://leetcode.com/problems/3sum/	1. two pointers: start = 1 and end 2. nums[start] + nums[end] + num 3. sorted the list first	Array	Yes	
Container With Most Water	https://leetcode.com/problems/container-with-most-water/	1. left and right pointer 2. area.append(min(height[left], h	Array	Yes	
Sum of Two Integers	https://leetcode.com/problems/sum-of-two-integers/	sum [a,b]	Binary	Yes	
Number of 1 Bits	https://leetcode.com/problems/number-of-1-bits/	bin(n).count("1")	Binary	Yes	
Counting Bits	https://leetcode.com/problems/counting-bits/	while x > 0: bit += x % 2 x = x//2	Binary	Yes	
Missing Number	https://leetcode.com/problems/missing-number/	https://www.geeksforgeeks.org/fir	Binary	Yes	
Reverse Bits	https://leetcode.com/problems/reverse-bits/	oribin='{0:032b}'.format(n) reversebin=oribin[::-1] return int(reversebin,2)	Binary	Yes	
Merge K Sorted Lists	https://leetcode.com/problems/merge-k-sorted-lists/		LinkedList	No	
Remove Nth Node From End of Li	https://leetcode.com/problems/remove-nth-node-from-end-of-list/	1. for loop to set the faster pointe 2. if not fast: return head.next NEED TO CHECK IF FAST.NEXT	LinkedList	Yes	
Reorder List	https://leetcode.com/problems/reorder-list/	1. Find middle of the linkedlist 2. reverse the later half of the list 3. merge two list	LinkedList	Yes	Need more practice
Longest Substring Without Repeat	https://leetcode.com/problems/longest-substring-without-repeating-cha	1. Have start = -1 and have a dict 2. if word are in dictionary (appea 3. max(the longest and the curre	String	Yes	
Longest Repeating Character Rel	https://leetcode.com/problems/longest-repeating-character-replacemer	https://leetcode.com/problems/lor	String	Yes	
Minimum Window Substring	https://leetcode.com/problems/minimum-window-substring/		String	No	
Valid Anagram	https://leetcode.com/problems/valid-anagram/	1. collection.counter 2. dictionary to keep track of elem	String	yes	
Group Anagrams	https://leetcode.com/problems/group-anagrams/	1. sorted every word and use dict	String	Yes	
Valid Parenteses	https://leetcode.com/problems/valid-parentheses/	1. while {}[]() in word: 2. s = s.replace({}, '') 1. stack https://github.com/neetcode-gh/le	String	Yes	

Valid Palindrome	https://leetcode.com/problems/valid-palindrome/	1. clean string isalnum 2. return s == s[::-1]	String	Yes					
Longest Palindromic Substring	https://leetcode.com/problems/longest-palindromic-substring/	1. need a helper function that che 2. you need to check for both odc	String	Yes					

Palindromic Substrings	https://leetcode.com/problems/palindromic-substrings/	for x in range(0, len(s)):
		for y in range(1, calc + 1):
		if s[x:x+y] == s[x:x+y][::
		res += 1
		calc -= 1
		return res
			String	Yes					

Encode and Decode String	https://leetcode.com/problems/encode-and-decode-strings/		String	Premium					
Maximum Depth of Binary Tree	https://leetcode.com/problems/maximum-depth-of-binary-tree/	dfs(root): if root == None: return 0 left = dfs(root.left) right = dfs(root.right) return 1+max(left,right)	Tree	Yes					

Same Tree	https://leetcode.com/problems/same-tree/	if p == None and q == None:
		return True
		if p == None or q == None:
		return False
		if p.val != q.val:
		return False
			Tree	Yes					

Invert/Flip Binary Tree	https://leetcode.com/problems/invert-binary-tree/	if root == None: return root root.left, root.right = root.righ root.left = self.invertTree(roo root.right = self.invertTree(rc	Tree	Yes					
Binary Tree Maximum Path Sum	https://leetcode.com/problems/binary-tree-maximum-path-sum/		Tree	No					
Binary Tree Level Order Traversal	https://leetcode.com/problems/binary-tree-level-order-traversal/	1. uses queue data structure. 2. BFS 3. initialize queue with root 4. inport deque https://leetcode.com/problems/bir	Tree	Yes					
Serialize and Deserialize Binary T	https://leetcode.com/problems/serialize-and-deserialize-binary-tree/		Tree	No					
Subtree of Another Tree	https://leetcode.com/problems/subtree-of-another-tree/		Tree	No					
Construct Binary Tree ferom Preor	https://leetcode.com/problems/construct-binary-tree-from-preorder-and-inorder-traversal/		Tree	No					

Validate Binary Search Tree	https://leetcode.com/problems/validate-binary-search-tree/	def valid(node, left, right):
		if node == None:
		return True
		
		if not left < node.val < righ
		return False
		return valid(node.left, left,
			Tree	Yes					

Kth Smallest Element in a BST	https://leetcode.com/problems/kth-smallest-element-in-a-bst/	1. Inorder traversal and append t 2. find the k-1 element	Tree	Yes					
Lowest Common Ancestor of BST	https://leetcode.com/problems/lowest-common-ancestor-of-a-binary-se	1. Binary search for tree 2. if root.val > p.val and root.val > root = root.left elif root.val < p.val and ro root = root.right	Tree	Yes					
Implement Trie (Prefix Tree)	https://leetcode.com/problems/implement-trie-prefix-tree/		Tree	No					
Add and Search Word	https://leetcode.com/problems/add-and-search-word-data-structure-design/		Tree	No					
Word Search II	https://leetcode.com/problems/word-search-ii/		Tree	No					
Clone Graph	https://leetcode.com/problems/clone-graph/		Graph	No					
Course Schedule	https://leetcode.com/problems/course-schedule/		Graph	No					
Pacific Atlantic Water Flow	https://leetcode.com/problems/pacific-atlantic-water-flow/		Graph	No					
Number of Islands	https://leetcode.com/problems/number-of-islands/	1. DFS function, if the cur island i 2. check boundaries 3. count += 1	Graph	Yes					

Longest Consecutive Sequuence	https://leetcode.com/problems/longest-consecutive-sequence/	1. set to filter out all the duplicate 2. sorted the list 3. compare index 1 to index 0 + 1 4. max(cur longest and max long	Graph	Yes
Allien Dictionary	https://leetcode.com/problems/alien-dictionary/		Graph	Premium
Graph Valid Tree	https://leetcode.com/problems/graph-valid-tree/		Graph	Premium
Number of Connected Componen	https://leetcode.com/problems/number-of-connected-components-in-an-undirected-graph/		Graph	Premium

Insert Interval	https://leetcode.com/problems/insert-interval/	intervals.append(newInterval)
		intervals = sorted(intervals)
		res = []
		for i in intervals:
		if res == [] or res[-1][1] < i
		res.append(i)
		else:
		res[-1][1] = max(res[-1]
		return res
			Interval	Yes

Merge Intervals	https://leetcode.com/problems/merge-intervals/	https://leetcode.com/problems/me	Interval	Yes
Non-overlapping Intervals	https://leetcode.com/problems/non-overlapping-intervals/	1. compare the 2nd element with 2. if the interval is valid. count +=	Interval	Yes
Meeting Rooms	https://leetcode.com/problems/meeting-rooms/		Interval	Premium
Meeting Rooms II	https://leetcode.com/problems/meeting-rooms-ii/		Interval	Premium
Set Matrix Zeros	https://leetcode.com/problems/set-matrix-zeroes/	1. do matrix traversal and store th 2. do another matrix traversal, no	Matrix	Yes
Spiral Matrix	https://leetcode.com/problems/spiral-matrix/		Matrix	No
Rotate IImage	https://leetcode.com/problems/rotate-image/		Matrix	No
Word Search	https://leetcode.com/problems/word-search/		Matrix	No
Merge K Sorted Lists	https://leetcode.com/problems/merge-k-sorted-lists/		Heap	No
Top K Frequent Elements	https://leetcode.com/problems/top-k-frequent-elements/		Heap	No
Find Median from Data Stream	https://leetcode.com/problems/find-median-from-data-stream/		Heap	No
Climbing Stairs	https://leetcode.com/problems/climbing-stairs/		Dynamic Prograr	No
Coin Change	https://leetcode.com/problems/coin-change/		Dynamic Prograr	No
Longest Increasing Subsequence	https://leetcode.com/problems/longest-increasing-subsequence/		Dynamic Prograr	No
Longest Common Subsequence	https://leetcode.com/problems/longest-common-subsequence/		Dynamic Prograr	No
Word Break Problem	https://leetcode.com/problems/word-break/		Dynamic Prograr	No
Combination Sum	https://leetcode.com/problems/combination-sum-iv/		Dynamic Prograr	No
House Robber	https://leetcode.com/problems/house-robber/		Dynamic Prograr	No
House Robber II	https://leetcode.com/problems/house-robber-ii/		Dynamic Prograr	No
Decode ways	https://leetcode.com/problems/decode-ways/		Dynamic Prograr	No
Unique Paths	https://leetcode.com/problems/unique-paths/		Dynamic Prograr	No
Jump Game	https://leetcode.com/problems/jump-game/		Dynamic Prograr	No