| | | # | Might be able to solve without looking at the solution | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Group | Tag | # | | Acceptance | Difficulty | | | | | | | | | |
| N-Sum | hashtable | 1 | Two Sum | | | | | | | | | | | |
| | two pointers, hashtable | 532 | K-diff Pairs in an Array | | | | | | | | | | | |
| | two pointers | 167 | Two Sum II - Input array is sorted | | | | | | | | | | | |
| | hashtable | 170 | Two Sum III - Data structure design | | | | | | | | | | | |
| | two pointers, hashtable | 15 | 3Sum | | | | | | | | | | | |
| | two pointers | 16 | 3Sum Closest | | | | | | | | | | | |
| | two pointers | 259 | 3Sum Smaller | | | | | | | | | | | |
| | two pointers | 611 | Valid Triangle Number | | | | | | | | | | | |
| | two pointers, hashtable | 18 | 4Sum | | | | | | | | | | | |
| | hashtable | 454 | 4Sum II | | | | | | | | | | | |
| Fixed Length Sliding Window | queue | 346 | Moving Average from Data Stream | | | | | | | | | | | |
| | two pointer | 643 | Maximum Average Subarray I | | | | | | | | | | | |
| | heap, deque | 239 | Sliding Window Maximum | | | | | | | | | | | |
| | heap, median | 295 | Find Median from Data Stream | | | | | | | | | | | |
| | heap, median | 480 | Sliding Window Median | | | | | | | | | | | |
| | two pointers, hashmap | 438 | Find All Anagrams in a String | | | | | | | | | | | |
| | two pointers, hashmap | 567 | Permutation in String | | | | | | | | | | | |
| | two pointers, hashmap | 30 | Substring with Concatenation of All Words | | | | | | | | | | | |
| variant length sliding window | two pointers, hashmap | 3 | Longest Substring Without Repeating Characters | | | | | | | | | | | |
| | regular expression | 395 | Longest Substring with At Least K Repeating Characters | | | | | | | | | | | |
| | two pointers | 159 | Longest Substring with At Most Two Distinct Characters | | | | | | | | | | | |
| | two pointers | 340 | Longest Substring with At Most K Distinct Characters | | | | | | | | | | | |
| | two pointers | 424 | Longest Repeating Character Replacement | | | | | | | | | | | |
| | two pointers, subsequence | 524 | Longest Word in Dictionary through Deleting | | | | | | | | | | | |
| | two pointers | 76 | Minimum Window Substring | | | | | | | | | | | |
| | two pointers | 209 | Minimum Size Subarray Sum | | | | | | | | | | | |
| subarray sum | hashmap | 325 | Maximum Size Subarray Sum Equals k | | | | | | | | | | | |
| | hashmap | 525 | Contiguous Array | | | | | | | | | | | |
| | hashmap | 523 | Continuous Subarray Sum | | | | | | | | | | | |
| | hashmap | 560 | Subarray Sum Equals K | | | | | | | | | | | |
| Array Organization | two pointers | 26 | Remove Duplicates from Sorted Array | | | | | | | | | | | |
| | two pointers | 80 | Remove Duplicates from Sorted Array II | | | | | | | | | | | |
| | two pointers | 27 | Remove Element | | | | | | | | | | | |
| | two pointers | 283 | Move Zeroes | | | | | | | | | | | |
| | two pointers | 88 | Merge Sorted Array | | | | | | | | | | | |
| | pointers, counting | 75 | Sort Colors | | | | | | | | | | | |
| Water Catch | two pointers | 11 | Container With Most Water | | | | | | | | | | | |
| | two pointers | 42 | Trapping Rain Water | | | | | | | | | | | |
| | stack | 84 | Largest Rectangle in Histogram | | | | | | | | | | | |
| | stack | 85 | Maximal Rectangle | | | | | | | | | | | |
| | heap | 407 | Trapping Rain Water II | | | | | | | | | | | |
| Circle | fast and slow | 141 | Linked List Cycle | | | | | | | | | | | |
| | fast and slow | 142 | Linked List Cycle II | | | | | | | | | | | |
| | fast and slow | 287 | Find the Duplicate Number | | n(logn) by binary search, or O(n) by fast and slow two pointers | | | | | | | | | |
| | | 160 | Intersection of Two Linked Lists | | can solve elegently by connenct AB and BA | | | | | | | | | |
| Remove Node from Linked List | | 237 | Delete Node in a Linked List | | do not need to change every node | | | | | | | | | |
| | | 83 | Remove Duplicates from Sorted List | | | | | | | | | | | |
| | | 82 | Remove Duplicates from Sorted List II | | | | | | | | | | | |
| | | 203 | Remove Linked List Elements | | | | | | | | | | | |
| | | 19 | Remove Nth Node From End of List | | | | | | | | | | | |
| move nodes | | 61 | Rotate Lists | | corner case: k = 0, < 0, =m * len(list) | | | | | | | | | |
| | | 86 | Partition List | | | | | | | | | | | |
| | | 328 | **Odd Even Linked List** | | can have short, elegent solution | | | | | | | | | |
| | | 206 | Reverse Linked List | | dummy.next, node.next, node = node, dummy.next, node.next<br>every time, from the old node, we remove head into the newhead, update both head and newhead | | | | | | | | | |
| | | 92 | Reverse Linked List II | | keep, pre and m-th, insert (m+1) to n nodes between pre and pre.next | | | | | | | | | |
| | | 21 | Merge Two Sorted Lists | | | | | | | | | | | |
| | | 23 | Merge k Sorted Lists | | | | | | | | | | | |

| | | | | | | |
|---|---|---|---|---|---|---|
| | | 147 | Insertion Sort List | | | |
| | fast and slow | 148 | Sort List | | | fast slow pointers, pay attantion to the fast and slow start position |
| | fast and slow | 143 | Reorder List | | | find right part, reverse right part, insert right into left |
| | | 24 | Swap Nodes in Pairs | | | recursive should be easier, while for iterative, we each time reverse two, and move two position |
| | | 25 | Reverse Nodes in k-Group | | | recursive |
| reverse array | | 344 | Reverse String | | | |
| | | 541 | Reverse String II | | | |
| | | 345 | Reverse Vowels of a String | | | |
| | | 151 | Reverse Words in a String | | | |
| | | 186 | Reverse Words in a String II | | | |
| | | 557 | Reverse Words in a String III | | | |
| | | 189 | Rotate Array | | | nums = nums[::-1] will not change the nums passed as parameter, but nums[:] = nums[::-1] will do |
| | | 7 | Reverse Integer | | | |
| | bit | 190 | Reverse Bits | | | get the bin in each position, and set the result correspondingly |
| tradictional BT | backtracking | 17 | Letter Combinations of a Phone Number | | | |
| | backtracking | 401 | Binary Watch | | | |
| | backtracking | 22 | Generate Parentheses | | | |
| | backtracking | 78 | Subsets | | | |
| | backtracking | 90 | Subsets II | | | |
| | backtracking | 491 | Increasing Subsequences | | | |
| | backtracking | 46 | Permutations | | | |
| | backtracking | 47 | Permutations II | | | |
| | backtracking | 77 | Combinations | | | |
| | backtracking | 39 | Combination Sum | | | |
| | backtracking | 40 | Combination Sum II | | | |
| | backtracking | 216 | Combination Sum III | | | |
| | namic programmi | 377 | Combination Sum IV | | | |
| | backtracking | 638 | Shopping Offers | | | |
| | backtracking | 254 | Factor Combinations | | | |
| Two Pointers | | | | | | |
| Palindrome | | 5 | Longest Palindromic Substring | 25.10% | Medium | |
| | | 647 | Palindromic Substrings | 56.00% | Medium | extension of 5, to find the longest, in fact we checked all the palindromic substrings |
| | | 9 | Palindrome Number | 34.70% | Easy | corner case, x< 0, x =0 and x is ***0 |
| | | 125 | Valid Palindrome | 25.90% | Easy | string.ascii_letters, digits, str.isalnum() |
| | | 680 | Valid Palindrome II | 28.10% | Easy | |
| | **KMP** | 214 | Shortest Palindrome | 23.60% | Hard | only s.startswith can pass, while s[:n-i] == r[i:] will TLE |
| | | 409 | Longest Palindrome | 45.10% | Easy | insert or remove from set |
| | | 266 | Palindrome Permutation | 56.30% | Easy | same as 409, use a set to add odd char and remove it when even (meet again) |
| | | 267 | Palindrome Permutation II | 31.50% | Medium | first determine whether can form as permuation, then if the odd exist, it must be in the middle<br>The left problem is the distint permutation of the even chars using backtracking |
| | | 234 | Palindrome Linked List | 32.20% | Easy | Two pointer to find the middle, and reverse the right part |
| | | 564 | Find the Closest Palindrome | 14.00% | Hard | |
| Binary Search | | | | | | |
| | | 35 | Search Insert Position | 39.40% | Easy | bisect.bisect_left |
| | | 34 | Search for a Range | 31.10% | Medium | bisect |
| | | 162 | Find Peak Element | 36.70% | Medium | |
| | | 74 | Search a 2D Matrix | 35.30% | Medium | |
| | | 240 | Search a 2D Matrix II | 38.10% | Medium | |
| Rotated Sorted Array | | 33 | Search in Rotated Sorted Array | 32.10% | Medium | |
| | | 81 | Search in Rotated Sorted Array II | 32.80% | Medium | When we find left == mid or right == mid, we have to move it towards mid until a different value found |
| | | 153 | Find Minimum in Rotated Sorted Array | 39.30% | Medium | |
| | | 154 | Find Minimum in Rotated Sorted Array II | 36.70% | Hard | |
| | | 540 | Single Element in a Sorted Array | 53.50% | Medium | |
| | | 4 | Median of Two Sorted Arrays | 21.30% | Hard | **Median of K sorted Arrays** |
| | | 315 | Count of Smaller Numbers After Self | 34.10% | Hard | binary search |
| | | 374 | Guess Number Higher or Lower | 34.50% | Easy | |
| | | 658 | Find K Closest Elements | 35.30% | Medium | |
| | | 278 | First Bad Version | 24.80% | Easy | |
| | | 302 | Smallest Rectangle Enclosing Black Pixels | 44.80% | Hard | |
| | | 441 | Arranging Coins | 36.10% | Easy | |
| | | 475 | Heaters | 29.60% | Easy | sort... and two pointers |
| Backtracking | | | | | | |
| Games.. | | 36 | Valid Sudoku | 34.90% | Medium | |
| | | 37 | Sudoku Solver | 29.20% | Hard | |
| | | 488 | Zuma Game | 36.20% | Hard | |
| | | 51 | N-Queens | 30.00% | Hard | |

| | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 52 | N-Queens II | 43.80% | Hard | | | | | | | | |
| | | 351 | Android Unlock Patterns | 43.20% | Medium | | | | | | | | |
| ab to 12 | string | 205 | Isomorphic Strings | 33.20% | Easy | | | | | | | | |
| | string | 290 | Word Pattern | 32.60% | Easy | | | | | | | | |
| | string | 291 | Word Pattern II | 37.80% | Hard | backtracking to test different length | | | | | | | |
| | string | 468 | Validate IP Address | 20.20% | Medium | | | | | | | | |
| | string | 93 | Restore IP Addresses | 26.60% | Medium | | | | | | | | |
| | string | 526 | Beautiful Arrangement | 54.10% | Medium | | | | | | | | |
| | string | 131 | Palindrome Partitioning | 32.00% | Medium | | | | | | | | |
| | string | 132 | Palindrome Partitioning II | 23.80% | Hard | dp | | | | | | | |
| | | 79 | Word Search | 26.10% | Medium | can early return instead of set a flag | | | | | | | |
| | | 212 | Word Search II | 22.90% | Hard | | | | | | | | |
| | | 127 | Word Ladder | 19.30% | Medium | | | | | | | | |
| | | 126 | Word Ladder II | 13.90% | Hard | | | | | | | | |
| | | 422 | Valid Word Square | 36.30% | Easy | | | | | | | | |
| | | 425 | Word Squares | 42.60% | Hard | | | | | | | | |
| abbreviation | | 408 | Valid Word Abbreviation | 27.60% | Easy | | | | | | | | |
| | | 288 | Unique Word Abbreviation | 16.10% | Medium | hashtable | | | | | | | |
| | | 320 | Generalized Abbreviation | 44.30% | Medium | | | | | | | | |
| | | 411 | Minimum Unique Word Abbreviation | 31.80% | Hard | brute force? | | | | | | | |
| | | 527 | Word Abbreviation | 34.80% | Hard | not a bt | | | | | | | |
| | | 282 | Expression Add Operators | 29.30% | Hard | | | | | | | | |
| | | 679 | 24 Game | 38.60% | Hard | | | | | | | | |
| Dynamic Programming (try to save space) | | | | | | | | | | | | | |
| | | 53 | Maximum Subarray | 39.20% | Easy | | | | | | | | |
| | | 152 | Maximum Product Subarray | 25.10% | Medium | | | | | | | | |
| | traverse | 238 | Product of Array Except Self | 48.20% | Medium | two pass, pay attention to the index in each side | | | | | | | |
| | traverse | 581 | Shortest Unsorted Continuous Subarray | 28.30% | Easy | | | | | | | | |
| | | 303 | Range Sum Query - Immutable | 28.00% | Easy | | | | | | | | |
| | | 304 | Range Sum Query 2D - Immutable | 24.00% | Medium | range matters | | | | | | | |
| | | 121 | Best Time to Buy and Sell Stock | 40.30% | Easy | | | | | | | | |
| stock dp | | 122 | Best Time to Buy and Sell Stock II | 46.30% | Easy | | | | | | | | |
| | | 123 | Best Time to Buy and Sell Stock III | 28.80% | Hard | There is a one pass dp using O(1) space | | | | | | | |
| | | 188 | Best Time to Buy and Sell Stock IV | 24.10% | Hard | Need to think clearly how the dp works | | | | | | | |
| | | 309 | Best Time to Buy and Sell Stock with Cooldown | 40.20% | Medium | | | | | | | | |
| O(n) DP, can do O(1) Space | | 70 | Climbing Stairs | 39.30% | Easy | sn = sn-1 + sn-2, s0 = 1, s1 = 0 | | | | | | | |
| | | 91 | Decode Ways | 19.30% | Medium | a lot of corner cases | | | | | | | |
| | | 639 | Decode Ways II | 18.90% | Hard | | | | | | | | |
| | | 198 | House Robber | 38.20% | Easy | | | | | | | | |
| | | 213 | House Robber II | 33.50% | Medium | | | | | | | | |
| | | 337 | House Robber III | 42.60% | Medium | indeed can do a postorder with another stack | | | | | | | |
| | | 256 | Paint House | 45.90% | Easy | | | | | | | | |
| | | 265 | Paint House II | 37.70% | Hard | | | | | | | | |
| | | 276 | Paint Fence | 34.20% | Easy | | | | | | | | |
| O(n) DP but need to check all possible position before | | 494 | Target Sum | 43.80% | Medium | | | | | | | | |
| | | 300 | Longest Increasing Subsequence | 37.90% | Medium | have a O(nlogn) solution | | | | | | | |
| | | 673 | Number of Longest Increasing Subsequence | 30.80% | Medium | | | | | | | | |
| | | 96 | Unique Binary Search Trees | 40.40% | Medium | | | | | | | | |
| | | 95 | Unique Binary Search Trees II | 31.00% | Medium | | | | | | | | |
| | | 279 | Perfect Squares | 36.00% | Medium | | | | | | | | |
| | | 322 | Coin Change | 26.20% | Medium | | | | | | | | |
| | | 343 | Integer Break | 45.50% | Medium | | | | | | | | |
| | | 139 | Word Break | 29.20% | Medium | | | | | | | | |
| | | 140 | Word Break II | 22.70% | Hard | | | | | | | | |
| M*N path O(M) space | | 120 | Triangle | 33.10% | Medium | | | | | | | | |
| | | 62 | Unique Paths | 40.20% | Medium | O(min(m, n)) space, best performance if we dp on the short direction | | | | | | | |
| | | 63 | Unique Paths II | 31.30% | Medium | | | | | | | | |
| | | 562 | Longest Line of Consecutive One in Matrix | 36.30% | Medium | | | | | | | | |
| | | 221 | Maximal Square | 28.00% | Medium | | | | | | | | |
| | | 64 | Minimum Path Sum | 37.80% | Medium | | | | | | | | |
| | | 174 | Dungeon Game | 23.40% | Hard | | | | | | | | |
| | | 568 | Maximum Vacation Days | 40.20% | Hard | | | | | | | | |
| | | 551 | Student Attendance Record I | 44.20% | Easy | | | | | | | | |
| | | 552 | Student Attendance Record II | 27.70% | Hard | N * (2*3) | | | | | | | |
| M*N path M*N Space | | 361 | Bomb Enemy | 38.60% | Medium | | | | | | | | |

| | | | | | | |
|---|---|---|---|---|---|---|
| String Matching (M+1)*(N+1) | | 44 | Wildcard Matching | 19.60% | Hard | might have a quick solution |
| | | 10 | Regular Expression Matching | 23.90% | Hard | can do a standard dp of (M+1) * (N+1) |
| | | 72 | Edit Distance | 31.10% | Hard | can do O(M) space |
| | | 87 | Scramble String | 28.70% | Hard | |
| | | 97 | Interleaving String | 24.30% | Hard | |
| | | 583 | Delete Operation for Two Strings | 40.80% | Medium | |
| | | 115 | Distinct Subsequences | 31.10% | Hard | can do O(M) space |
| | | 89 | Gray Code | 40.30% | Medium | no need to do a standard bt, but just extend the result, similar to subset |
| | | 375 | Guess Number Higher or Lower II | 35.60% | Medium | |
| | | 629 | K Inverse Pairs Array | 15.90% | Medium | |
| | | 600 | Non-negative Integers without Consecutive Ones | 21.60% | Hard | |
| | | 678 | Valid Parenthesis String | 26.20% | Medium | |
| | | 241 | Different Ways to Add Parentheses | 42.70% | Medium | |
| | | 312 | Burst Balloons | 42.20% | Hard | |
| | | 321 | Create Maximum Number | 24.30% | Hard | |
| | | 354 | Russian Doll Envelopes | 32.00% | Hard | |
| | | 363 | Max Sum of Rectangle No Larger Than K | 32.50% | Hard | |
| | | 368 | Largest Divisible Subset | 33.50% | Medium | |
| | | 410 | Split Array Largest Sum | 35.20% | Hard | |
| | | 416 | Partition Equal Subset Sum | 38.50% | Medium | |
| | | 403 | Frog Jump | 31.60% | Hard | |
| | | 293 | Flip Game | 54.90% | Easy | |
| | | 294 | Flip Game II | 45.90% | Medium | |
| | | 464 | Can I Win | 23.80% | Medium | |
| | | 486 | Predict the Winner | 44.40% | Medium | |
| | | 413 | Arithmetic Slices | 54.90% | Medium | |
| | | 446 | Arithmetic Slices II - Subsequence | 25.30% | Hard | |
| | | 467 | Unique Substrings in Wraparound String | 31.40% | Medium | |
| | | 466 | Count The Repetitions | 26.60% | Hard | |
| | | 472 | Concatenated Words | 29.70% | Hard | |
| | | 474 | Ones and Zeroes | 37.70% | Medium | |
| | | 514 | Freedom Trail | 34.90% | Hard | |
| | | 516 | Longest Palindromic Subsequence | 42.40% | Medium | |
| | | 517 | Super Washing Machines | 35.60% | Hard | |
| | | 546 | Remove Boxes | 29.60% | Hard | |
| | | 553 | Optimal Division | 53.70% | Medium | |
| | | 471 | Encode String with Shortest Length | 41.60% | Hard | |
| | | 576 | Out of Boundary Paths | 0.324 | Hard | |
| Design | | | | | | |
| list | | 380 | Insert Delete GetRandom O(1) | 38.90% | Medium | list for random O(1), dict for insert and remove O(1) |
| | | 381 | Insert Delete GetRandom O(1) - Duplicates allowed | 28.50% | Hard | |
| linked list | | 146 | LRU Cache | 17.00% | Hard | double linked list for get max, min, most recent O(1), need to update both direction |
| | | 460 | LFU Cache | 22.40% | Hard | double linked list with ordered dict |
| | | 432 | All O`one Data Structure | 27.50% | Hard | |
| stack | | 155 | Min Stack | 27.60% | Easy | |
| | | 225 | Implement Stack using Queues | 32.10% | Easy | |
| | | 232 | Implement Queue using Stacks | 35.90% | Easy | |
| Iterator | | 251 | Flatten 2D Vector | 39.80% | Medium | |
| | | 281 | Zigzag Iterator | 49.60% | Medium | |
| | | 284 | Peeking Iterator | 35.30% | Medium | |
| | | 341 | Flatten Nested List Iterator | 40.30% | Medium | |
| | | 604 | Design Compressed String Iterator | 30.20% | Easy | |
| | | 353 | Design Snake Game | 26.10% | Medium | |
| | | 379 | Design Phone Directory | 31.40% | Medium | |
| | hashtable | 631 | Design Excel Sum Formula | 17.00% | Hard | |
| | hashtable | 359 | Logger Rate Limiter | 59.00% | Easy | |
| | | 635 | Design Log Storage System | 42.60% | Medium | |
| | | 362 | Design Hit Counter | 53.30% | Medium | |
| | trie | 208 | Implement Trie (Prefix Tree) | 27.00% | Medium | dict is faster than a list in OJ |
| | trie | 211 | Add and Search Word - Data structure design | 21.50% | Medium | |
| | trie | 677 | Map Sum Pairs | 54.00% | Medium | |
| | trie | 642 | Design Search Autocomplete System | 26.70% | Hard | |
| | | 676 | Implement Magic Dictionary | 51.10% | Medium | |
| | hashtable | 535 | Encode and Decode TinyURL | 74.60% | Medium | |
| | | 271 | Encode and Decode Strings | 26.20% | Medium | |
| | | 355 | Design Twitter | 25.00% | Medium | |

| | | | | | | |
|---|---|---|---|---|---|---|
| | | 348 | Design Tic-Tac-Toe | 45.60% | Medium | |
| | | 588 | Design In-Memory File System | 30.90% | Hard | |
| Tree | | | | | | |
| Traversal | inorder | 94 | Binary Tree Inorder Traversal | 45.30% | Medium | iterative, Morris traversal |
| | preorder | 144 | Binary Tree Preorder Traversal | 44.10% | Medium | iterative |
| | postorder | 145 | Binary Tree Postorder Traversal | 39.40% | Hard | iterative, stack with status, inorder and preorder can be solved similarly |
| | inorder | 173 | Binary Search Tree Iterator | 40.30% | Medium | |
| | inorder | 98 | Validate Binary Search Tree | 22.90% | Medium | |
| | inorder | 99 | Recover Binary Search Tree | 29.30% | Hard | |
| | inorder | 285 | Inorder Successor in BST | 36.00% | Medium | |
| | preorder | 606 | Construct String from Binary Tree | 54.90% | Easy | |
| | inorder | 105 | Construct Binary Tree from Preorder and Inorder Traversal | 31.50% | Medium | hashmap might make it faster |
| | inorder | 106 | Construct Binary Tree from Inorder and Postorder Traversal | 31.50% | Medium | |
| | preorder | 255 | Verify Preorder Sequence in Binary Search Tree | 39.60% | Medium | |
| | inorder | 501 | Find Mode in Binary Search Tree | 38.40% | Easy | |
| | inorder | 653 | Two Sum IV - Input is a BST | 50.50% | Easy | |
| Level Order Traversal (mostly BFS) | | 102 | Binary Tree Level Order Traversal | 38.40% | Medium | |
| | | 107 | Binary Tree Level Order Traversal II | 39.10% | Easy | |
| | | 339 | Nested List Weight Sum | 60.90% | Easy | |
| | | 364 | Nested List Weight Sum II | 51.40% | Medium | |
| | | 103 | Binary Tree Zigzag Level Order Traversal | 33.50% | Medium | |
| | | 515 | Find Largest Value in Each Tree Row | 53.90% | Medium | |
| | | 116 | Populating Next Right Pointers in Each Node | 36.90% | Medium | we need to set next, not left, right |
| | | 117 | Populating Next Right Pointers in Each Node II | 33.60% | Medium | |
| | | 199 | Binary Tree Right Side View | 39.80% | Medium | |
| | | 623 | Add One Row to Tree | 50.10% | Medium | |
| | | 637 | Average of Levels in Binary Tree | 63.20% | Easy | |
| BFS/DFS (iterative/recursive) | postorder | 110 | Balanced Binary Tree | 36.90% | Easy | |
| | | 100 | Same Tree | 45.90% | Easy | |
| | | 101 | Symmetric Tree | 37.90% | Easy | |
| | | 226 | Invert Binary Tree | 50.90% | Easy | recursive, iterative (dfs using stack and bfs using deque, appendleft and pop) |
| | | 617 | Merge Two Binary Trees | 73.40% | Easy | |
| | | 314 | Binary Tree Vertical Order Traversal | 36.20% | Medium | |
| | | 104 | Maximum Depth of Binary Tree | 51.80% | Easy | |
| | DFS | 563 | Binary Tree Tilt | 49.00% | Easy | |
| | | 111 | Minimum Depth of Binary Tree | 32.70% | Easy | |
| | | 257 | Binary Tree Paths | 36.90% | Easy | |
| | | 404 | Sum of Left Leaves | 46.60% | Easy | |
| | DFS | 112 | Path Sum | 33.50% | Easy | 48.96 iterative |
| | DFS | 113 | Path Sum II | 32.60% | Medium | iterative using stack and extra status |
| | BFS/DFS | 437 | Path Sum III | 39.30% | Easy | a dfs method with 98.87 |
| | DFS | 124 | Binary Tree Maximum Path Sum | 25.50% | Hard | |
| | BFS/DFS | 513 | Find Bottom Left Tree Value | 55.80% | Medium | |
| | BFS/DFS | 129 | Sum Root to Leaf Numbers | 35.90% | Medium | |
| | BFS/DFS | 298 | Binary Tree Longest Consecutive Sequence | 40.50% | Medium | |
| | DFS | 250 | Count Univalue Subtrees | 41.20% | Medium | |
| | left and right | 549 | Binary Tree Longest Consecutive Sequence II | 36.80% | Medium | two different writing have different performance, saving almost half time |
| | left and right | 543 | Diameter of Binary Tree | 42.80% | Easy | the diameter is not the number of nodes |
| | | 366 | Find Leaves of Binary Tree | 58.60% | Medium | |
| | | 310 | Minimum Height Trees | 28.70% | Medium | |
| | | 508 | Most Frequent Subtree Sum | 52.00% | Medium | |
| | inorder | 538 | Convert BST to Greater Tree | 52.90% | Medium | |
| | | 156 | Binary Tree Upside Down | 43.60% | Medium | |
| | | 545 | Boundary of Binary Tree | 28.20% | Medium | |
| Tree Conversion | stack | 114 | Flatten Binary Tree to Linked List | 34.30% | Medium | |
| | stack | 536 | Construct Binary Tree from String | 38.30% | Medium | |
| | binary | 108 | Convert Sorted Array to Binary Search Tree | 41.40% | Easy | |
| | binary | 109 | Convert Sorted List to Binary Search Tree | 33.40% | Medium | |
| Serialization | | 572 | Subtree of Another Tree | 43.90% | Easy | serialize, recursive, iterative |
| | | 331 | Verify Preorder Serialization of a Binary Tree | 35.70% | Medium | |
| | stack | 297 | Serialize and Deserialize Binary Tree | 32.60% | Hard | preorder, recover with a stack. or BFS with a list |
| | | 449 | Serialize and Deserialize BST | 42.20% | Medium | |
| | binary search | 222 | Count Complete Tree Nodes | 27.10% | Medium | |
| | hashtable | 133 | Clone Graph | 25.10% | Medium | |
| | hashtable | 138 | Copy List with Random Pointer | 26.50% | Medium | O(1) extra space solution |
| | adjacency list | 323 | Number of Connected Components in an Undirected Graph | 47.50% | Medium | |

| | | | | | | |
|---|---|---|---|---|---|---|
| | adjacency list | 261 | Graph Valid Tree | 37.30% | Medium | |
| Binary Search Tree | | 230 | Kth Smallest Element in a BST | 43.00% | Medium | |
| | | 235 | Lowest Common Ancestor of a Binary Search Tree | 38.50% | Easy | |
| | | 236 | Lowest Common Ancestor of a Binary Tree | 29.60% | Medium | |
| | | 270 | Closest Binary Search Tree Value | 38.90% | Easy | |
| | | 272 | Closest Binary Search Tree Value II | 38.40% | Hard | |
| | postorder | 333 | Largest BST Subtree | 30.20% | Medium | |
| | | 450 | Delete Node in a BST | 35.80% | Medium | |
| | | 530 | Minimum Absolute Difference in BST | 47.50% | Easy | |
| | | 669 | Trim a Binary Search Tree | 59.60% | Easy | |
| Topological Sort | | 207 | Course Schedule | 31.30% | Medium | |
| | | 210 | Course Schedule II | 26.80% | Medium | |
| | | 444 | Sequence Reconstruction | 19.70% | Medium | |
| | traverse | 269 | Alien Dictionary | 22.80% | Hard | |
| | | 329 | Longest Increasing Path in a Matrix | 35.90% | Hard | |
| BFS | | 582 | Kill Process | 42.50% | Medium | |
| BFS | | 130 | Surrounded Regions | 17.90% | Medium | |
| BFS | | 200 | Number of Islands | 33.60% | Medium | |
| BFS | | 305 | Number of Islands II | 38.60% | Hard | |
| BFS | | 286 | Walls and Gates | 43.50% | Medium | |
| | | 490 | The Maze | 42.40% | Medium | |
| | | 505 | The Maze II | 36.10% | Medium | dijkstra.. Heap |
| | | 499 | The Maze III | 31.00% | Hard | |
| | | 301 | Remove Invalid Parentheses | 34.90% | Hard | |
| | | 317 | Shortest Distance from All Buildings | 33.60% | Hard | |
| DFS | | 332 | Reconstruct Itinerary | 28.70% | Medium | |
| | | 399 | Evaluate Division | 40.30% | Medium | |
| DFS | | 547 | Friend Circles | 49.00% | Medium | |
| BFS | | 417 | Pacific Atlantic Water Flow | 33.20% | Medium | |
| BFS | | 419 | Battleships in a Board | 61.20% | Medium | |
| BFS | | 473 | Matchsticks to Square | 34.30% | Medium | |
| BFS | | 529 | Minesweeper | 51.80% | Medium | |
| BFS on all | | 542 | 01 Matrix | 32.40% | Medium | |
| Segment Tree | | | | | | |
| | | 307 | Range Sum Query - Mutable | 19.60% | Medium | |
| | | 308 | Range Sum Query 2D - Mutable | 21.50% | Hard | |
| | | 327 | Count of Range Sum | 29.30% | Hard | |
| | | 493 | Reverse Pairs | 19.10% | Hard | |
| Greedy | | | | | | |
| | | 561 | Array Partition I | 73.20% | Easy | |
| | | 55 | Jump Game | 29.30% | Medium | |
| | | 45 | Jump Game II | 26.10% | Hard | |
| | | 455 | Assign Cookies | 47.20% | Easy | |
| | | 134 | Gas Station | 29.00% | Medium | |
| | | 392 | Is Subsequence | 44.30% | Medium | |
| | | 630 | Course Schedule III | 8.90% | Medium | |
| | | 406 | Queue Reconstruction by Height | 54.70% | Medium | |
| | | 418 | Sentence Screen Fitting | 27.40% | Medium | |
| | | 484 | Find Permutation | 52.30% | Medium | |
| | | 452 | Minimum Number of Arrows to Burst Balloons | 43.40% | Medium | |
| | | 135 | Candy | 24.30% | Hard | |
| | | 316 | Remove Duplicate Letters | 29.10% | Hard | |
| | | 330 | Patching Array | 31.70% | Hard | |
| | | 621 | Task Scheduler | 38.60% | Medium | |
| | | 358 | Rearrange String k Distance Apart | 31.80% | Hard | |
| | | 68 | Text Justification | 18.60% | Hard | |
| | | 605 | Can Place Flowers | 29.20% | Easy | |
| Bit Manipulation | | | | | | |
| | | 136 | Single Number | 53.70% | Easy | 60.79 using defaultdict |
| | | 137 | Single Number II | 40.80% | Medium | 63.24 using defaultdict |
| | | 260 | Single Number III | 50.50% | Medium | 86.24 using defaultdict |
| | | 371 | Sum of Two Integers | 51.20% | Easy | Have not idea how to solve it... |
| | | 191 | Number of 1 Bits | 39.10% | Easy | |
| | | 338 | Counting Bits | 60.40% | Medium | |
| | | 476 | Number Complement | 61.30% | Easy | 101 to 010 can be done by 101 ^ 111 |
| | | 461 | Hamming Distance | 70.70% | Easy | |

| | | | | | | |
|---|---|---|---|---|---|---|
| | | 477 | Total Hamming Distance | 46.40% | Medium | |
| | | 201 | Bitwise AND of Numbers Range | 33.50% | Medium | |
| | | 318 | Maximum Product of Word Lengths | 43.30% | Medium | |
| | | 397 | Integer Replacement | 29.60% | Medium | |
| | | 421 | Maximum XOR of Two Numbers in an Array | 44.50% | Medium | |
| | | | Array | | | |
| | | | String | | | |
| | | | Math1 | | | |
| ADD | | 2 | Add Two Numbers | 27.20% | Medium | Append node into the current list |
| | | 445 | Add Two Numbers II | 46.10% | Medium | stack, insert after dummy to form the correct order |
| | | 67 | Add Binary | 31.50% | Easy | consider each list separately as a list, in other words, solve the problem as 2 |
| | | 66 | Plus One | 37.80% | Easy | Can to similar as 445, while a briliant way is to determine whether the current digit is 9, if not, then we do not have carry and can return early |
| | | 369 | Plus One Linked List | 53.90% | Medium | Either reverse the list using a stack as 445, or can do a recursion so that for each next node, we return whether it has a carry |
| | | 415 | Add Strings | 41.20% | Easy | Similar to 2, do not forget to append the carry, and reverse the result |
| Multiplication | | 43 | Multiply Strings | 26.50% | Medium | It is more convenient to have a direct index/pointer for the array, and reduce the index each time than calculate the index |
| | | 537 | Complex Number Multiplication | 65.40% | Medium | |
| | | 311 | Sparse Matrix Multiplication | 50.60% | Medium | |
| Divide | | 29 | Divide Two Integers | 16.00% | Medium | log(N) binary search to find all possible divisor * factor that is less than dividend. Dividend / divisor = quotient and remainder |
| | | 166 | Fraction to Recurring Decimal | 17.20% | Medium | Corner Cases: numerator is 0, denominator is 0, negative number and sign |
| Square | | 69 | Sqrt(x) | 27.50% | Easy | binary search, corner case is mid*mid <= x, but (mid+1)**2 > x, then mid is the result |
| | | 367 | Valid Perfect Square | 37.90% | Easy | standard binary search. 1+3+5+(2n-1) = ... |
| | | 633 | Sum of Square Numbers | 27.80% | Easy | |
| Power | | 50 | Pow(x, n) | 26.70% | Medium | Same as sqrt, but x**2 will overflow while x*x will not<br>There are lots of different solutions. A quick solution is recursion, pow(x, n) = pow(x*x, n/2) * (1 if n % 2 else x) |
| | | 372 | Super Pow | 33.80% | Medium | |
| | | 231 | Power of Two | 39.80% | Easy | Count 1 in binary format, or bit manipulation x & x-1 should be 0 |
| | | 326 | Power of Three | 39.80% | Easy | log10(n)/log10(3) % 1 == 0 |
| | | 342 | Power of Four | 38.00% | Easy | same as power of two, but need to remove those are power of 2 but not 4 |
| String | | 8 | String to Integer (atoi) | 13.90% | Medium | need to pay attention to corner cases, like max/min int, the idx |
| | | 65 | Valid Number | 12.70% | Hard | corner cases, while idx < n and (a or b) |
| | | 12 | Integer to Roman | 43.80% | Medium | just consider each case... |
| | | 13 | Roman to Integer | 44.80% | Easy | same to 12 |
| | | 273 | Integer to English Words | 21.70% | Hard | corner case: 0 |
| Base | | 168 | Excel Sheet Column Title | 25.30% | Easy | idx is n-1 % 26 |
| | | 171 | Excel Sheet Column Number | 46.20% | Easy | |
| | | 405 | Convert a Number to Hexadecimal | 40.80% | Easy | if num < 0:<br>num = num + 2**32 |
| | | 504 | Base 7 | 45.50% | Easy | |
| | | 660 | Remove 9 | 44.70% | Hard | |
| Traverse an arry or matrix | | 6 | ZigZag Conversion | 26.40% | Medium | |
| | | 498 | Diagonal Traverse | 46.40% | Medium | row+col=s, row in [0, m-1] thus col in [0 or s-m+1, s or n -1] |
| | | 48 | Rotate Image | 37.90% | Medium | |
| | | 54 | Spiral Matrix | 25.30% | Medium | |
| | | 59 | Spiral Matrix II | 38.80% | Medium | |
| | | 73 | Set Matrix Zeroes | 35.50% | Medium | |
| | | 661 | Image Smoother | 47.90% | Easy | |
| | | 520 | Detect Capital | 52.50% | Easy | |
| | | 616 | Add Bold Tag in String | 36.20% | Medium | |
| | | 289 | Game of Life | 36.50% | Medium | |
| | | 118 | Pascal's Triangle | 37.70% | Easy | |
| | | 119 | Pascal's Triangle II | 35.90% | Easy | |
| | | 566 | Reshape the Matrix | 67.90% | Easy | |
| | | 391 | Perfect Rectangle | 25.60% | Hard | |
| | | 382 | Linked List Random Node | 46.60% | Medium | |
| | | 398 | Random Pick Index | 41.80% | Medium | |
| | | 384 | Shuffle an Array | 45.90% | Medium | |
| | | 624 | Maximum Distance in Arrays | 31.00% | Easy | |
| | | 628 | Maximum Product of Three Numbers | 44.20% | Easy | |
| | | 657 | Judge Route Circle | 71.90% | Easy | str.count |
| | | 674 | Longest Continuous Increasing Subsequence | 44.00% | Easy | |
| | | 20 | Valid Parentheses | 32.90% | Easy | |
| | | 32 | Longest Valid Parentheses | 23.00% | Hard | |
| | | 224 | Basic Calculator | 26.30% | Hard | |
| | | 227 | Basic Calculator II | 28.70% | Medium | |
| | | 71 | Simplify Path | 24.70% | Medium | |
| | | 388 | Longest Absolute File Path | 35.90% | Medium | |

| | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Stack | | 150 | Evaluate Reverse Polish Notation | 26.60% | Medium | | | | | | | | |
| | | 591 | Tag Validator | 17.90% | Hard | corner cases | | | | | | | |
| | | 385 | Mini Parser | 30.00% | Medium | | | | | | | | |
| | | 439 | Ternary Expression Parser | 50.30% | Medium | | | | | | | | |
| | | 394 | Decode String | 40.90% | Medium | | | | | | | | |
| | | 496 | Next Greater Element I | 57.60% | Easy | | | | | | | | |
| | | 503 | Next Greater Element II | 47.20% | Medium | | | | | | | | |
| | | 556 | Next Greater Element III | 27.20% | Medium | | | | | | | | |
| | | 218 | The Skyline Problem | 26.60% | Hard | | | | | | | | |
| missing or extra number | traverse | 41 | First Missing Positive | 25.20% | Hard | mark negative to 0, and turn value in exisiting index to negative, finally the first positive position is missing | | | | | | | |
| | traverse | 268 | Missing Number | 44.00% | Easy | 1+..+n = (n+1)*n/2 | | | | | | | |
| | traverse | 169 | Majority Element | 45.80% | Easy | | | | | | | | |
| | traverse | 229 | Majority Element II | 28.20% | Medium | | | | | | | | |
| | hashtable | 217 | Contains Duplicate | 44.80% | Easy | set | | | | | | | |
| | hashtable | 219 | Contains Duplicate II | 32.00% | Easy | hashtable to store the last idx | | | | | | | |
| | hashtable | 220 | Contains Duplicate III | 19.20% | Medium | | | | | | | | |
| | traverse | 442 | Find All Duplicates in an Array | 53.80% | Medium | add n is a good idea | | | | | | | |
| | traverse | 448 | Find All Numbers Disappeared in an Array | 52.50% | Easy | | | | | | | | |
| | hashtable | 389 | Find the Difference | 51.50% | Easy | | | | | | | | |
| | hashtable | 645 | Set Mismatch | 40.60% | Easy | | | | | | | | |
| Interval | traverse | 56 | Merge Intervals | 29.30% | Medium | | | | | | | | |
| | traverse | 57 | Insert Interval | 27.00% | Hard | | | | | | | | |
| | traverse | 252 | Meeting Rooms | 46.50% | Easy | | | | | | | | |
| | greedy | 253 | Meeting Rooms II | 38.70% | Medium | | | | | | | | |
| | binary search | 436 | Find Right Interval | 41.20% | Medium | | | | | | | | |
| | | 435 | Non-overlapping Intervals | 40.40% | Medium | greedy | | | | | | | |
| | traverse | 495 | Teemo Attacking | 52.00% | Medium | | | | | | | | |
| | traverse | 163 | Missing Ranges | 25.50% | Medium | | | | | | | | |
| | traverse | 228 | Summary Ranges | 28.90% | Medium | | | | | | | | |
| | | 352 | Data Stream as Disjoint Intervals | 39.50% | Hard | | | | | | | | |
| Number | hashtable | 202 | Happy Number | 40.00% | Easy | | | | | | | | |
| | math | 258 | Add Digits | 50.70% | Easy | | | | | | | | |
| | math | 507 | Perfect Number | 32.90% | Easy | | | | | | | | |
| | traverse | 306 | Additive Number | 27.30% | Medium | | | | | | | | |
| | traverse | 38 | Count and Say | 33.60% | Easy | | | | | | | | |
| | hashtable | 204 | Count Primes | 26.40% | Easy | corner case, and how to reduce duplicate multiply | | | | | | | |
| | math | 246 | Strobogrammatic Number | 39.40% | Easy | | | | | | | | |
| | math | 247 | Strobogrammatic Number II | 39.10% | Medium | | | | | | | | |
| | math | 248 | Strobogrammatic Number III | 31.20% | Hard | DP | | | | | | | |
| | math | 263 | Ugly Number | 38.80% | Easy | <=0 | | | | | | | |
| | math | 264 | Ugly Number II | 32.10% | Medium | | | | | | | | |
| | math | 313 | Super Ugly Number | 37.30% | Medium | | | | | | | | |
| | math | 172 | Factorial Trailing Zeroes | 35.50% | Easy | | | | | | | | |
| | math | 625 | Minimum Factorization | 27.70% | Medium | | | | | | | | |
| | | 479 | Largest Palindrome Product | 17.90% | Easy | | | | | | | | |
| | math | 357 | Count Numbers with Unique Digits | 45.60% | Medium | | | | | | | | |
| | math | 233 | Number of Digit One | 27.90% | Hard | | | | | | | | |
| | math | 396 | Rotate Function | 31.50% | Medium | | | | | | | | |
| | math | 483 | Smallest Good Base | 31.40% | Hard | | | | | | | | |
| | math | 453 | Minimum Moves to Equal Array Elements | 46.80% | Easy | min | | | | | | | |
| | math | 462 | Minimum Moves to Equal Array Elements II | 51.30% | Medium | median | | | | | | | |
| | math | 296 | Best Meeting Point | 51.30% | Hard | median | | | | | | | |
| | math | 573 | Squirrel Simulation | 46.70% | Medium | | | | | | | | |
| Digital or String order | traverse | 31 | Next Permutation | 28.50% | Medium | | | | | | | | |
| | | 386 | Lexicographical Numbers | 40.50% | Medium | | | | | | | | |
| | | 60 | Permutation Sequence | 27.80% | Medium | | | | | | | | |
| | | 440 | K-th Smallest in Lexicographical Order | 23.30% | Hard | | | | | | | | |
| | | 555 | Split Concatenated Strings | 28.50% | Medium | | | | | | | | |
| | | 400 | Nth Digit | 30.10% | Easy | | | | | | | | |
| | | 179 | Largest Number | 22.10% | Medium | | | | | | | | |
| | stack | 402 | Remove K Digits | 26.10% | Medium | | | | | | | | |
| Top N | traverse | 414 | Third Maximum Number | 27.50% | Easy | corner cases | | | | | | | |
| | heap | 378 | Kth Smallest Element in a Sorted Matrix | 43.80% | Medium | can also binary search | | | | | | | |
| | heap | 373 | Find K Pairs with Smallest Sums | 30.40% | Medium | | | | | | | | |
| | heap | 215 | Kth Largest Element in an Array | 38.40% | Medium | | | | | | | | |

| | | | | | | |
|---|---|---|---|---|---|---|
| | heap | 347 | Top K Frequent Elements | 47.30% | Medium | bucket<br>LFU/LRU |
| | | 632 | Smallest Range | 43.50% | Hard | |
| | heap | 659 | Split Array into Consecutive Subsequences | 33.20% | Medium | |
| | heap | 502 | IPO | 34.70% | Hard | |
| | bucket | 539 | Minimum Time Difference | 45.50% | Medium | |
| | bucket | 128 | Longest Consecutive Sequence | 36.00% | Hard | can solve use hashmap too |
| | bucket | 164 | Maximum Gap | 29.10% | Hard | |
| Shape | | 463 | Island Perimeter | 56.80% | Easy | |
| | | 492 | Construct the Rectangle | 49.10% | Easy | |
| | math | 223 | Rectangle Area | 32.40% | Medium | |
| | math | 335 | Self Crossing | 24.70% | Hard | |
| | | 593 | Valid Square | 36.10% | Medium | |
| | math | 469 | Convex Polygon | 30.60% | Medium | |
| | math | 587 | Erect the Fence | 0.226 | Hard | |
| Logitic Game | | 292 | Nim Game | 55.10% | Easy | 1,2,3 win, 4 loss, 5,6,7 win, 8 loss |
| | math | 319 | Bulb Switcher | 42.20% | Medium | |
| | math | 672 | Bulb Switcher II | 45.60% | Medium | |
| 132 Pattern | dp | 376 | Wiggle Subsequence | 35.20% | Medium | |
| | | 280 | Wiggle Sort | 56.20% | Medium | |
| | | 324 | Wiggle Sort II | 25.50% | Medium | |
| | | 334 | Increasing Triplet Subsequence | 38.60% | Medium | |
| | | 456 | 132 Pattern | 28.20% | Medium | |
| | | 360 | Sort Transformed Array | 43.60% | Medium | |
| | | 634 | Find the Derangement of An Array | 26.60% | Medium | |
| | traverse | 370 | Range Addition | 54.60% | Medium | |
| | | 598 | Range Addition II | 44.40% | Easy | tricky.. |
| | | 640 | Solve the Equation | 38.60% | Medium | |
| | | 544 | Output Contest Matches | 72.10% | Medium | |
| | | 565 | Array Nesting | 47.10% | Medium | |
| | traverse | 390 | Elimination Game | 40.50% | Medium | binary |
| | traverse | 420 | Strong Password Checker | 20.10% | Hard | |
| | traverse | 393 | UTF-8 Validation | 34.80% | Medium | do int comparison instead of string |
| | traverse | 459 | Repeated Substring Pattern | 38.50% | Easy | |
| | traverse | 465 | Optimal Account Balancing | 34.00% | Hard | |
| | traverse | 481 | Magical String | 45.20% | Medium | |
| | traverse | 506 | Relative Ranks | 47.40% | Easy | |
| | traverse | 521 | Longest Uncommon Subsequence I | 51.30% | Easy | |
| | traverse | 522 | Longest Uncommon Subsequence II | 28.70% | Medium | |
| | traverse | 531 | Lonely Pixel I | 51.60% | Medium | |
| | traverse | 533 | Lonely Pixel II | 40.00% | Medium | |
| | traverse | 548 | Split Array with Equal Sum | 29.30% | Medium | |
| | traverse | 277 | Find the Celebrity | 35.30% | Medium | |
| | traverse | 482 | License Key Formatting | 41.20% | Medium | replace and upper is faster than do each position |
| | traverse | 412 | Fizz Buzz | 58.80% | Easy | |
| | traverse | 14 | Longest Common Prefix | 31.10% | Easy | |
| | traverse, KMP | 28 | Implement strStr() | 27.60% | Easy | |
| | traverse | 161 | One Edit Distance | 30.90% | Medium | |
| | traverse | 58 | Length of Last Word | 31.50% | Easy | |
| | traverse | 434 | Number of Segments in a String | 37.00% | Easy | |
| | traverse | 485 | Max Consecutive Ones | 54.50% | Easy | |
| | traverse | 487 | Max Consecutive Ones II | 44.40% | Medium | Follow Up: if we can flip at most K zero, we store the previous at most K zero index in a queue, so that we can calculate the len when new value come in. Queue is better |
| | traverse | 157 | Read N Characters Given Read4 | 29.10% | Easy | |
| | traverse | 158 | Read N Characters Given Read4 II - Call multiple times | 24.30% | Hard | |
| | traverse | 165 | Compare Version Numbers | 19.70% | Medium | |
| Hash Table | | | | | | |
| | | 349 | Intersection of Two Arrays | 46.60% | Easy | set.intersection(t) |
| | | 350 | Intersection of Two Arrays II | 44.30% | Easy | (countera & counterb).elements() |
| | | 599 | Minimum Index Sum of Two Lists | 57.50% | Easy | |
| | | 299 | Bulls and Cows | 34.00% | Medium | |
| | | 49 | Group Anagrams | 33.30% | Medium | count sort may make the str sort quicker.. |
| | | 242 | Valid Anagram | 45.70% | Easy | hashmap is faster than len_26 char |
| | | 383 | Ransom Note | 46.70% | Easy | |
| | | 447 | Number of Boomerangs | 44.10% | Easy | |
| | | 356 | Line Reflection | 30.20% | Medium | |
| | gcd | 592 | Fraction Addition and Subtraction | 46.80% | Medium | gcd, and re |

| | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | gcd | 365 | Water and Jug Problem | 26.70% | Medium | | | | | | | | |
| | gcd | 149 | Max Points on a Line | 15.40% | Hard | | | | | | | | |
| | | 249 | Group Shifted Strings | 40.30% | Medium | | | | | | | | |
| | | 187 | Repeated DNA Sequences | 30.60% | Medium | | | | | | | | |
| word distance not edit distance | travese | 243 | Shortest Word Distance | 51.60% | Easy | just traverse the word list, update the distance, the initial set is important | | | | | | | |
| | | 244 | Shortest Word Distance II | 36.30% | Medium | | | | | | | | |
| | travese | 245 | Shortest Word Distance III | 49.80% | Medium | only need to slightly modify 243 | | | | | | | |
| | | 387 | First Unique Character in a String | 46.40% | Easy | | | | | | | | |
| | | 594 | Longest Harmonious Subsequence | 36.20% | Easy | | | | | | | | |
| | | 423 | Reconstruct Original Digits from English | 43.20% | Medium | | | | | | | | |
| | | 451 | Sort Characters By Frequency | 50.60% | Medium | | | | | | | | |
| | | 500 | Keyboard Row | 60.30% | Easy | | | | | | | | |
| | | 575 | Distribute Candies | 0.648 | Easy | | | | | | | | |
| | | 554 | Brick Wall | 41.60% | Medium | | | | | | | | |
| | | 274 | H-Index | 32.70% | Medium | | | | | | | | |
| | | 275 | H-Index II | 33.90% | Medium | | | | | | | | |
| | | 609 | Find Duplicate File in System | 54.80% | Medium | | | | | | | | |
| | | 336 | Palindrome Pairs | 25.60% | Hard | | | | | | | | |