

CRACKING THE PLACEMENT TESTS

A WEEKLY GUIDE CONTAINING TOPICS, QUESTIONS AND TRICKS COMMONLY ENCOUNTERED IN PLACEMENT TESTS

July 11, 2020

CODING CLUB, IIT GUWAHATI

SORTING

* VERY IMPORTANT TOPIC FOR PLACEMENTS. VERY IMPORTANT TO HAVE A FIRM GRIP ON ITS CONCEPTS.

READING MATERIALS

* [Sorting Terminology](https://www.geeksforgeeks.org/sorting-terminology/)
* [Stability in sorting algorithms](https://www.geeksforgeeks.org/stability-in-sorting-algorithms/)
* [Time complexities of Standard Sorting Algorithms](https://www.geeksforgeeks.org/time-complexities-of-all-sorting-algorithms/)
* [Selection Sort](https://www.geeksforgeeks.org/selection-sort/)
* [Bubble Sort](https://www.geeksforgeeks.org/bubble-sort/)
* [Insertion Sort](https://www.geeksforgeeks.org/insertion-sort/)
* [Merge Sort](https://www.geeksforgeeks.org/merge-sort/)
* [Quick Sort](https://www.geeksforgeeks.org/quick-sort/)
* [Heap Sort](https://www.geeksforgeeks.org/heap-sort/)
* [Counting Sort](https://www.geeksforgeeks.org/counting-sort/)
* [Radix Sort](https://www.geeksforgeeks.org/radix-sort/)
* [Bucket Sort](https://www.geeksforgeeks.org/bucket-sort-2/)
* [Shell Sort](https://www.geeksforgeeks.org/shellsort/)
* [External Sorting](https://www.geeksforgeeks.org/external-sorting/)

We generally have inbuilt sorting functions in the library. This makes it easier when it comes to coding. For details, refer to the articles below

1. [Know Your Sorting Algorithm | Set 1 (Sorting Weapons used by Programming Languages)](https://www.geeksforgeeks.org/know-sorting-algorithm-set-1-sorting-weapons-used-programming-languages/)
2. [Know Your Sorting Algorithm | Set 2 (Introsort- C++’s Sorting Weapon)](https://www.geeksforgeeks.org/know-your-sorting-algorithm-set-2-introsort-cs-sorting-weapon/)
3. [Comparator function of qsort() in C](https://www.geeksforgeeks.org/comparator-function-of-qsort-in-c/)
4. [sort() in C++ STL](https://www.geeksforgeeks.org/sort-c-stl/)
5. [C qsort() vs C++ sort()](https://www.geeksforgeeks.org/c-qsort-vs-c-sort/)

PRACTICE

1. cses.fi [THIS CONTAINS QUESTIONS ON BOTH SORTING AND SEARCHING]

* [Sorting Methods](https://cses.fi/problemset/task/1162)
* [Distinct Numbers](https://cses.fi/problemset/task/1621)
* [Apartments](https://cses.fi/problemset/task/1084)
* [Ferris Wheel](https://cses.fi/problemset/task/1090)
* [Concert Tickets](https://cses.fi/problemset/task/1091)
* [Restaurant Customers](https://cses.fi/problemset/task/1619)
* [Movie Festival](https://cses.fi/problemset/task/1629)
* [Sum of Two Values](https://cses.fi/problemset/task/1640)
* [Maximum Subarray Sum](https://cses.fi/problemset/task/1643)
* [Stick Lengths](https://cses.fi/problemset/task/1074)
* [Playlist](https://cses.fi/problemset/task/1141)
* [Towers](https://cses.fi/problemset/task/1073)
* [Traffic Lights](https://cses.fi/problemset/task/1163)
* [Room Allocation](https://cses.fi/problemset/task/1164)
* [Factory Machines](https://cses.fi/problemset/task/1620)
* [Tasks and Deadlines](https://cses.fi/problemset/task/1630)
* [Reading Books](https://cses.fi/problemset/task/1631)
* [Sum of Three Values](https://cses.fi/problemset/task/1641)
* [Sum of Four Values](https://cses.fi/problemset/task/1642)
* [Nearest Smaller Values](https://cses.fi/problemset/task/1645)
* [Subarray Sums I](https://cses.fi/problemset/task/1660)
* [Subarray Sums II](https://cses.fi/problemset/task/1661)
* [Subarray Divisibility](https://cses.fi/problemset/task/1662)
* [Array Division](https://cses.fi/problemset/task/1085)
* [Sliding Median](https://cses.fi/problemset/task/1076)
* [Sliding Cost](https://cses.fi/problemset/task/1077)
* [Movie Festival II](https://cses.fi/problemset/task/1632)
* [Maximum Subarray Sum II](https://cses.fi/problemset/task/1644)

1. InterviewBit

|  |
| --- |
| * [Noble Integer](https://www.interviewbit.com/problems/noble-integer/) |
| * [Wave Array](https://www.interviewbit.com/problems/wave-array/) |  |
| * [Hotel Bookings Possible](https://www.interviewbit.com/problems/hotel-bookings-possible/) |  |
| * [Maximum Unsorted Subarray](https://www.interviewbit.com/problems/maximum-unsorted-subarray/) |  |
| * [Max Distance](https://www.interviewbit.com/problems/max-distance/) |  |  |  |  |

1. GFG [ONLY SORTING]
2. [Sort elements by frequency | Set 1](https://www.geeksforgeeks.org/sort-elements-by-frequency/)
3. [Sort elements by frequency | Set 2](https://www.geeksforgeeks.org/sort-elements-by-frequency-set-2/)
4. [Count Inversions in an array | Set 1 (Using Merge Sort)](https://www.geeksforgeeks.org/counting-inversions/)
5. [Sort an array of 0s, 1s and 2s](https://www.geeksforgeeks.org/sort-an-array-of-0s-1s-and-2s/)
6. [Find the Minimum length Unsorted Subarray, sorting which makes the complete array sorted](https://www.geeksforgeeks.org/minimum-length-unsorted-subarray-sorting-which-makes-the-complete-array-sorted/)
7. [Find whether an array is subset of another array | Added Method 3](https://www.geeksforgeeks.org/find-whether-an-array-is-subset-of-another-array-set-1/)
8. [Sort a nearly sorted (or K sorted) array](https://www.geeksforgeeks.org/nearly-sorted-algorithm/)
9. [Sort numbers stored on different machines](https://www.geeksforgeeks.org/sort-numbers-stored-on-different-machines/)
10. [Sort a linked list of 0s, 1s and 2s](https://www.geeksforgeeks.org/sort-a-linked-list-of-0s-1s-or-2s/)
11. [A Pancake Sorting Problem](https://www.geeksforgeeks.org/a-pancake-sorting-question/)
12. [Find number of pairs (x, y) in an array such that x^y > y^x](https://www.geeksforgeeks.org/find-number-pairs-xy-yx/)
13. [Count all distinct pairs with difference equal to k](https://www.geeksforgeeks.org/count-pairs-difference-equal-k/)
14. [C Program for Bubble Sort on Linked List](https://www.geeksforgeeks.org/c-program-bubble-sort-linked-list/)
15. [Sort n numbers in range from 0 to n^2 – 1 in linear time](https://www.geeksforgeeks.org/sort-n-numbers-range-0-n2-1-linear-time/)
16. [C Program to Sort an array of names or strings](https://www.geeksforgeeks.org/c-program-sort-array-names-strings/)
17. [Sort an array according to the order defined by another array](https://www.geeksforgeeks.org/sort-array-according-order-defined-another-array/)
18. [Given a sorted array and a number x, find the pair in array whose sum is closest to x](https://www.geeksforgeeks.org/given-sorted-array-number-x-find-pair-array-whose-sum-closest-x/)
19. [Sort an array in wave form](https://www.geeksforgeeks.org/sort-array-wave-form-2/)
20. [Check if any two intervals overlap among a given set of intervals](https://www.geeksforgeeks.org/check-if-any-two-intervals-overlap-among-a-given-set-of-intervals/)
21. [How to efficiently sort a big list dates in 20’s](https://www.geeksforgeeks.org/how-to-efficiently-sort-a-big-list-dates-in-20s/)
22. [Sort an almost sorted array where only two elements are swapped](https://www.geeksforgeeks.org/sort-an-almost-sorted-array-where-only-two-elements-are-swapped/)
23. [Find the point where maximum intervals overlap](https://www.geeksforgeeks.org/find-the-point-where-maximum-intervals-overlap/)
24. [Sort a linked list that is sorted alternating ascending and descending orders?](https://www.geeksforgeeks.org/how-to-sort-a-linked-list-that-is-sorted-alternating-ascending-and-descending-orders/)
25. [C++ program for Sorting Dates using Selection Sort](https://www.geeksforgeeks.org/c-program-for-sorting-dates-using-selection-sort/)
26. [How to sort an array of dates in C/C++?](https://www.geeksforgeeks.org/how-to-sort-an-array-of-dates-in-cc/)
27. [Sorting Strings using Bubble Sort](https://www.geeksforgeeks.org/sorting-strings-using-bubble-sort-2/)
28. [Maximum product of a triplet (subsequnece of size 3) in array](https://www.geeksforgeeks.org/find-maximum-product-of-a-triplet-in-array/)
29. [Find missing elements of a range](https://www.geeksforgeeks.org/find-missing-elements-of-a-range/)
30. [Find a permutation that causes worst case of Merge Sort](https://www.geeksforgeeks.org/find-a-permutation-that-causes-worst-case-of-merge-sort/)
31. [Minimum sum of two numbers formed from digits of an array](https://www.geeksforgeeks.org/minimum-sum-two-numbers-formed-digits-array/)
32. [Find minimum difference between any two elements](https://www.geeksforgeeks.org/find-minimum-difference-pair/)
33. [Convert an array to reduced form | Set 1 (Simple and Hashing)](https://www.geeksforgeeks.org/convert-an-array-to-reduced-form-set-1-simple-and-hashing/)
34. [Sorting Vector of Pairs in C++ | Set 1 (Sort by first and second)](https://www.geeksforgeeks.org/sorting-vector-of-pairs-in-c-set-1-sort-by-first-and-second/)
35. [Sorting Vector of Pairs in C++ | Set 2 (Sort in descending order by first and second)](https://www.geeksforgeeks.org/sorting-vector-of-pairs-in-c-set-2-sort-in-descending-order-by-first-and-second/)
36. [Sorting 2D Vector in C++ | Set 1 (By row and column)](https://www.geeksforgeeks.org/sorting-2d-vector-in-c-set-1-by-row-and-column/)
37. [Sorting 2D Vector in C++ | Set 2 (In descending order by row and column)](https://www.geeksforgeeks.org/sorting-2d-vector-in-c-set-2-in-descending-order-by-row-and-column/)
38. [Sorting 2D Vector in C++ | Set 3 (By number of columns)](https://www.geeksforgeeks.org/sorting-2d-vector-c-set-3-number-columns/)
39. [Find Surpasser Count of each element in array](https://www.geeksforgeeks.org/find-surpasser-count-of-each-element-in-array/)
40. [Rearrange positive and negative numbers with constant extra space](https://www.geeksforgeeks.org/rearrange-positive-and-negative-numbers/)
41. [Sort an array according to count of set bits](https://www.geeksforgeeks.org/sort-array-according-count-set-bits/)
42. [Count distinct occurrences as a subsequence](https://www.geeksforgeeks.org/count-distinct-occurrences-as-a-subsequence/)
43. [Minimum number of swaps required to sort an array](https://www.geeksforgeeks.org/minimum-number-swaps-required-sort-array/)
44. [Number of swaps to sort when only adjacent swapping allowed](https://www.geeksforgeeks.org/number-swaps-sort-adjacent-swapping-allowed/)
45. [Minimum swaps to make two arrays identical](https://www.geeksforgeeks.org/minimum-swaps-to-make-two-array-identical/)
46. [Find elements larger than half of the elements in an array](https://www.geeksforgeeks.org/find-elements-larger-half-elements-array/)
47. [Count minimum number of subsets (or subsequences) with consecutive numbers](https://www.geeksforgeeks.org/count-minimum-number-subsets-subsequences-consecutive-numbers/)
48. [Sum of all elements between k1’th and k2’th smallest elements](https://www.geeksforgeeks.org/sum-elements-k1th-k2th-smallest-elements/)
49. [Number of sextuplets (or six values) that satisfy an equation](https://www.geeksforgeeks.org/number-sextuplets-six-values-satisfy-equation/)
50. [Sort an array according to absolute difference with given value](https://www.geeksforgeeks.org/sort-an-array-according-to-absolute-difference-with-given-value/)
51. [Minimize the sum of product of two arrays with permutations allowed](https://www.geeksforgeeks.org/minimize-sum-product-two-arrays-permutations-allowed/)
52. [Position of an element after stable sort](https://www.geeksforgeeks.org/position-element-stable-sort/)
53. [Chocolate Distribution Problem](https://www.geeksforgeeks.org/chocolate-distribution-problem/)
54. [Sort even-placed elements in increasing and odd-placed in decreasing order](https://www.geeksforgeeks.org/sort-even-placed-elements-increasing-odd-placed-decreasing-order/)
55. [Permute two arrays such that sum of every pair is greater or equal to K](https://www.geeksforgeeks.org/permute-two-arrays-sum-every-pair-greater-equal-k/)
56. [Chose k array elements such that difference of maximum and minimum is minimized](https://www.geeksforgeeks.org/k-numbers-difference-maximum-minimum-k-number-minimized/)
57. [Sort an array when two halves are sorted](https://www.geeksforgeeks.org/sort-array-two-halves-sorted/)
58. [Find pair with greatest product in array](https://www.geeksforgeeks.org/find-pair-with-greatest-product-in-array/)
59. [Minimum swap required to convert binary tree to binary search tree](https://www.geeksforgeeks.org/minimum-swap-required-convert-binary-tree-binary-search-tree/)
60. [K-th smallest element after removing some integers from natural numbers](https://www.geeksforgeeks.org/k-th-smallest-element-removing-integers-natural-numbers/)
61. [Check whether Arithmetic Progression can be formed from the given array](https://www.geeksforgeeks.org/check-whether-arithmetic-progression-can-formed-given-array/)
62. [Bucket Sort to Sort an Array with Negative Numbers](https://www.geeksforgeeks.org/bucket-sort-to-sort-an-array-with-negative-numbers/)
63. [Possible to form a triangle from array values](https://www.geeksforgeeks.org/possible-form-triangle-array-values/)
64. [Maximum difference between frequency of two elements such that element having greater frequency is also greater](https://www.geeksforgeeks.org/maximum-difference-between-frequency-of-two-elements-such-that-element-having-greater-frequency-is-also-greater/)
65. [Check if reversing a sub array make the array sorted](https://www.geeksforgeeks.org/check-reversing-sub-array-make-array-sorted/)
66. [Find all triplets with zero sum](https://www.geeksforgeeks.org/find-triplets-array-whose-sum-equal-zero/)
67. [Sort a Matrix in all way increasing order](https://www.geeksforgeeks.org/sort-matrix-way-increasing-order/)
68. [Sort array after converting elements to their squares](https://www.geeksforgeeks.org/sort-array-converting-elements-squares/)
69. [Sort all even numbers in ascending order and then sort all odd numbers in descending order](https://www.geeksforgeeks.org/sort-even-numbers-ascending-order-sort-odd-numbers-descending-order/)
70. [Sorting Big Integers](https://www.geeksforgeeks.org/sorting-big-integers/)
71. [Sort an array of large numbers](https://www.geeksforgeeks.org/sort-array-large-numbers/)
72. [Sort 3 Integers without using if condition or using only max() function](https://www.geeksforgeeks.org/sort-3-integers-without-using-condition-using-max-function/)
73. [Minimum difference between max and min of all K-size subsets](https://www.geeksforgeeks.org/minimum-difference-max-min-k-size-subsets/)
74. [Minimum swaps to reach permuted array with at most 2 positions left swaps allowed](https://www.geeksforgeeks.org/minimum-swaps-reach-permuted-array-2-positions-left-swaps-allowed/)
75. [Convert an array to reduced form | Set 2 (Using vector of pairs)](https://www.geeksforgeeks.org/convert-array-reduced-form-set-2-using-vector-pairs/)
76. [Find sum of non-repeating (distinct) elements in an array](https://www.geeksforgeeks.org/find-sum-non-repeating-distinct-elements-array/)
77. [Minimum sum of absolute difference of pairs of two arrays](https://www.geeksforgeeks.org/minimum-sum-absolute-difference-pairs-two-arrays/)
78. [Find the largest multiple of 3 from array of digits | Set 2 (In O(n) time and O(1) space)](https://www.geeksforgeeks.org/find-largest-multiple-3-array-digits-set-2-time-o1-space/)
79. [Noble integers in an array (count of greater elements is equal to value)](https://www.geeksforgeeks.org/noble-integers-in-an-array-count-of-greater-elements-is-equal-to-value/)
80. [Find maximum height pyramid from the given array of objects](https://www.geeksforgeeks.org/find-maximum-height-pyramid-from-the-given-array-of-objects/)
81. [Program to check if an array is sorted or not (Iterative and Recursive)](https://www.geeksforgeeks.org/program-check-array-sorted-not-iterative-recursive/)
82. [Smallest Difference Triplet from Three arrays](https://www.geeksforgeeks.org/smallest-difference-triplet-from-three-arrays/)
83. [Smallest Difference pair of values between two unsorted Arrays](https://www.geeksforgeeks.org/smallest-difference-pair-values-two-unsorted-arrays/)
84. [Find whether it is possible to make array elements same using one external number](https://www.geeksforgeeks.org/find-whether-possible-make-array-elements-using-one-external-number/)
85. [Sort an array of strings according to string lengths](https://www.geeksforgeeks.org/sort-array-strings-according-string-lengths/)
86. [Check if it is possible to sort an array with conditional swapping of adjacent allowed](https://www.geeksforgeeks.org/check-possible-sort-array-conditional-swapping-adjacent-allowed/)
87. [Sort an array after applying the given equation](https://www.geeksforgeeks.org/sort-array-applying-given-equation/)
88. [Print array of strings in sorted order without copying one string into another](https://www.geeksforgeeks.org/print-array-strings-sorted-order-without-copying-one-string-another/)
89. [Sort elements on the basis of number of factors](https://www.geeksforgeeks.org/sort-elements-basis-number-factors/)

FOR MORE: <https://leetcode.com/tag/sort/>

SEARCHING

Searching Algorithms are designed to check for an element or retrieve an element from any data structure where it is stored. Based on the type of search operation, these algorithms are generally classified into two categories:

1. **Sequential Search**: In this, the list or array is traversed sequentially and every element is checked. For example: [Linear Search](https://www.geeksforgeeks.org/linear-search/).
2. **Interval Search**: These algorithms are specifically designed for searching in sorted data-structures. These type of searching algorithms are much more efficient than Linear Search as they repeatedly target the center of the search structure and divide the search space in half. For Example: [Binary Search](https://www.geeksforgeeks.org/binary-search/).

* [Linear Search](https://www.geeksforgeeks.org/linear-search/)
* [Binary Search](https://www.geeksforgeeks.org/binary-search/)
* [Ternary Search](https://www.geeksforgeeks.org/ternary-search/)
* [Ternary Search - cp algorithms](https://cp-algorithms.com/num_methods/ternary_search.html)
* [Linear Search vs Binary Search](https://www.geeksforgeeks.org/linear-search-vs-binary-search/)
* [Why is Binary Search preferred over Ternary Search?](https://www.geeksforgeeks.org/binary-search-preferred-ternary-search/)
* [**Binary Search functions in C++ STL (binary\_search, lower\_bound and upper\_bound)**](https://www.geeksforgeeks.org/binary-search-functions-in-c-stl-binary_search-lower_bound-and-upper_bound/) **[IMPORTANT]**

[NOT SO IMPORTANT BUT MICROSOFT HAS ASKED THIS IN PREVIOUS CODING TESTS]

* [Jump Search](https://www.geeksforgeeks.org/jump-search/)
* [Interpolation Search](https://www.geeksforgeeks.org/interpolation-search/)
* [Exponential Search](https://www.geeksforgeeks.org/exponential-search/)

SEARCHING IS A VERY DIVERSE TOPIC AND IS PRESENT IN ALMOST EVERY TOPIC

e.g searching for a string in another string, searching in BST

BUT WE WILL COVER THESE UNDER THEIR RESPECTIVE HEADINGS

WE FOCUS MAINLY ON BINARY SEARCH AND TERNARY SEARCH HERE

ALSO, WE WILL LEARN ABOUT A TECHNIQUE CALLED

* [BINARY SEARCH THE ANSWER](https://oldaddr.wordpress.com/2014/06/28/binary-search-the-answer/)

SOME USEFUL LINKS:

* [Binary Search Tutorial (1) (Video)](https://www.youtube.com/watch?v=GU7DpgHINWQ)
* [Binary Search Tutorial (2)](https://www.topcoder.com/community/competitive-programming/tutorials/binary-search)
* [Binary\_search, Lower\_bound, Upper\_bound in C++](https://www.geeksforgeeks.org/binary-search-functions-in-c-stl-binary_search-lower_bound-and-upper_bound/)
* [Binary Search The Answer (Commonly known as BSTA)](https://oldaddr.wordpress.com/2014/06/28/binary-search-the-answer/)
* [One-Side Binary Search](https://www.geeksforgeeks.org/meta-binary-search-one-sided-binary-search/?ref=rp)
* [Ternary Search (Advanced)](https://cp-algorithms.com/num_methods/ternary_search.html)
* [Parallel Binary Search (Advanced)](https://codeforces.com/blog/entry/45578)

PRACTICE

<https://www.interviewbit.com/courses/programming/topics/binary-search/>

<https://www.codechef.com/LRNDSA04>

1. [Find the Missing Number](https://www.geeksforgeeks.org/find-the-missing-number/)
2. [Search an element in a sorted and rotated array](https://www.geeksforgeeks.org/search-an-element-in-a-sorted-and-pivoted-array/)
3. [Median of two sorted arrays](https://www.geeksforgeeks.org/median-of-two-sorted-arrays/)
4. [Two elements whose sum is closest to zero](https://www.geeksforgeeks.org/two-elements-whose-sum-is-closest-to-zero/)
5. [Find the smallest and second smallest element in an array](https://www.geeksforgeeks.org/to-find-smallest-and-second-smallest-element-in-an-array/)
6. [Maximum and minimum of an array using minimum number of comparisons](https://www.geeksforgeeks.org/maximum-and-minimum-in-an-array/)
7. [k largest(or smallest) elements in an array | added Min Heap method](https://www.geeksforgeeks.org/k-largestor-smallest-elements-in-an-array/)
8. [Ceiling in a sorted array](https://www.geeksforgeeks.org/ceiling-in-a-sorted-array/)
9. [Count number of occurrences (or frequency) in a sorted array](https://www.geeksforgeeks.org/count-number-of-occurrences-or-frequency-in-a-sorted-array/)
10. [Find the repeating and the missing | Added 3 new methods](https://www.geeksforgeeks.org/find-a-repeating-and-a-missing-number/)
11. [Find a Fixed Point in a given array](https://www.geeksforgeeks.org/find-a-fixed-point-in-a-given-array/)
12. [Find the maximum element in an array which is first increasing and then decreasing](https://www.geeksforgeeks.org/find-the-maximum-element-in-an-array-which-is-first-increasing-and-then-decreasing/)
13. [Find a pair with the given difference](https://www.geeksforgeeks.org/find-a-pair-with-the-given-difference/)
14. [Find the k most frequent words from a file](https://www.geeksforgeeks.org/find-the-k-most-frequent-words-from-a-file/)
15. [Median of two sorted arrays of different sizes](https://www.geeksforgeeks.org/median-of-two-sorted-arrays-of-different-sizes/)
16. [Find a peak element](https://www.geeksforgeeks.org/find-a-peak-in-a-given-array/)
17. [Given an array of of size n and a number k, find all elements that appear more than n/k times](https://www.geeksforgeeks.org/given-an-array-of-of-size-n-finds-all-the-elements-that-appear-more-than-nk-times/)
18. [Find the minimum element in a sorted and rotated array](https://www.geeksforgeeks.org/find-minimum-element-in-a-sorted-and-rotated-array/)
19. [Kth smallest element in a row-wise and column-wise sorted 2D array | Set 1](https://www.geeksforgeeks.org/kth-smallest-element-in-a-row-wise-and-column-wise-sorted-2d-array-set-1/)
20. [Find k closest elements to a given value](https://www.geeksforgeeks.org/find-k-closest-elements-given-value/)
21. [Search in an almost sorted array](https://www.geeksforgeeks.org/search-almost-sorted-array/)
22. [A Problem in Many Binary Search Implementations](https://www.geeksforgeeks.org/problem-binary-search-implementations/)
23. [Find the first repeating element in an array of integers](https://www.geeksforgeeks.org/find-first-repeating-element-array-integers/)
24. [Find common elements in three sorted arrays](https://www.geeksforgeeks.org/find-common-elements-three-sorted-arrays/)
25. [Count 1’s in a sorted binary array](https://www.geeksforgeeks.org/count-1s-sorted-binary-array/)
26. [Given a sorted array and a number x, find the pair in array whose sum is closest to x](https://www.geeksforgeeks.org/given-sorted-array-number-x-find-pair-array-whose-sum-closest-x/)
27. [Find the closest pair from two sorted arrays](https://www.geeksforgeeks.org/given-two-sorted-arrays-number-x-find-pair-whose-sum-closest-x/)
28. [K’th Smallest/Largest Element in Unsorted Array | Set 1](https://www.geeksforgeeks.org/kth-smallestlargest-element-unsorted-array/)
29. [K’th Smallest/Largest Element in Unsorted Array | Set 2 (Expected Linear Time)](https://www.geeksforgeeks.org/kth-smallestlargest-element-unsorted-array-set-2-expected-linear-time-2/)
30. [K’th Smallest/Largest Element in Unsorted Array | Set 3 (Worst Case Linear Time)](https://www.geeksforgeeks.org/kth-smallestlargest-element-unsorted-array-set-3-worst-case-linear-time/)
31. [Find position of an element in a sorted array of infinite numbers](https://www.geeksforgeeks.org/find-position-element-sorted-array-infinite-numbers/)
32. [Given a sorted and rotated array, find if there is a pair with a given sum](https://www.geeksforgeeks.org/given-a-sorted-and-rotated-array-find-if-there-is-a-pair-with-a-given-sum/)
33. [Find the largest pair sum in an unsorted array](https://www.geeksforgeeks.org/find-the-largest-pair-sum-in-an-unsorted-array/)
34. [Find the nearest smaller numbers on left side in an array](https://www.geeksforgeeks.org/find-the-nearest-smaller-numbers-on-left-side-in-an-array/)
35. [K’th largest element in a stream](https://www.geeksforgeeks.org/kth-largest-element-in-a-stream/)
36. [Find a pair with maximum product in array of Integers](https://www.geeksforgeeks.org/return-a-pair-with-maximum-product-in-array-of-integers/)
37. [Find the element that appears once in a sorted array](https://www.geeksforgeeks.org/find-the-element-that-appears-once-in-a-sorted-array/)
38. [Find the odd appearing element in O(Log n) time](https://www.geeksforgeeks.org/find-the-element-that-odd-number-of-times-in-olog-n-time/)
39. [Find the largest three elements in an array](https://www.geeksforgeeks.org/find-the-largest-three-elements-in-an-array/)
40. [Search an element in an array where difference between adjacent elements is 1](https://www.geeksforgeeks.org/search-an-element-in-an-array-where-difference-between-adjacent-elements-is-1/)
41. [Find three closest elements from given three sorted arrays](https://www.geeksforgeeks.org/find-three-closest-elements-from-given-three-sorted-arrays/)
42. [Find the element before which all the elements are smaller than it, and after which all are greater](https://www.geeksforgeeks.org/find-the-element-before-which-all-the-elements-are-smaller-than-it-and-after-which-all-are-greater-than-it/)
43. [Binary Search for Rational Numbers without using floating point arithmetic](https://www.geeksforgeeks.org/binary-search-for-rational-numbers-without-using-floating-point-arithmetic/)
44. [Floor in a Sorted Array](https://www.geeksforgeeks.org/floor-in-a-sorted-array/)
45. [Third largest element in an array of distinct elements](https://www.geeksforgeeks.org/third-largest-element-array-distinct-elements/)
46. [Second minimum element using minimum comparisons](https://www.geeksforgeeks.org/second-minimum-element-using-minimum-comparisons/)
47. [Queries for greater than and not less than](https://www.geeksforgeeks.org/queries-greater-not-less/)
48. [Efficient search in an array where difference between adjacent is 1](https://www.geeksforgeeks.org/efficient-search-in-an-array-where-difference-between-adjacent-is-1/)
49. [Print all possible sums of consecutive numbers with sum N](https://www.geeksforgeeks.org/print-possible-sums-consecutive-numbers-sum-n/)
50. [Minimum time required to produce m items](https://www.geeksforgeeks.org/minimum-time-required-produce-m-items/)
51. [Make all array elements equal with minimum cost](https://www.geeksforgeeks.org/make-array-elements-equal-minimum-cost/)
52. [Check if there exist two elements in an array whose sum is equal to the sum of rest of the array](https://www.geeksforgeeks.org/check-exist-two-elements-array-whose-sum-equal-sum-rest-array/)
53. [Check if reversing a sub array make the array sorted](https://www.geeksforgeeks.org/check-reversing-sub-array-make-array-sorted/)
54. [Find all triplets with zero sum](https://www.geeksforgeeks.org/find-triplets-array-whose-sum-equal-zero/)
55. [Search, insert and delete in an unsorted array](https://www.geeksforgeeks.org/search-insert-and-delete-in-an-unsorted-array/)
56. [Search, insert and delete in a sorted array](https://www.geeksforgeeks.org/search-insert-and-delete-in-a-sorted-array/)
57. [Move all occurrences of an element to end in a linked list](https://www.geeksforgeeks.org/move-occurrences-element-end-linked-list/)
58. [Search in an array of strings where non-empty strings are sorted](https://www.geeksforgeeks.org/search-in-an-array-of-strings-where-non-empty-strings-are-sorted/)
59. [Smallest Difference Triplet from Three arrays](https://www.geeksforgeeks.org/smallest-difference-triplet-from-three-arrays/)
60. [Best First Search (Informed Search)](https://www.geeksforgeeks.org/best-first-search-informed-search/)

RECURSION & BACKTRACKING

<https://www.hackerearth.com/practice/basic-programming/recursion/recursion-and-backtracking/tutorial/>

[Introduction to Backtracking - Brute Force Approach](https://www.youtube.com/watch?v=DKCbsiDBN6c)

[N Queens Problem using Backtracking](https://www.youtube.com/watch?v=xFv_Hl4B83A)

[Sum Of Subsets Problem - Backtracking](https://www.youtube.com/watch?v=kyLxTdsT8ws)

PRACTICE

<https://www.interviewbit.com/courses/programming/topics/backtracking/>

[**GCD Strings**](https://www.hackerearth.com/practice/basic-programming/recursion/recursion-and-backtracking/practice-problems/algorithm/gcd-strings/)

[**Lockdown Game**](https://www.hackerearth.com/practice/basic-programming/recursion/recursion-and-backtracking/practice-problems/algorithm/lockdown-game/)

[**Hack the money**](https://www.hackerearth.com/practice/basic-programming/recursion/recursion-and-backtracking/practice-problems/algorithm/hack-the-money/)

[**A Tryst With Chess**](https://www.hackerearth.com/practice/basic-programming/recursion/recursion-and-backtracking/practice-problems/algorithm/a-tryst-with-chess/)

[**Its Confidential**](https://www.hackerearth.com/practice/basic-programming/recursion/recursion-and-backtracking/practice-problems/algorithm/its-confidential-f006e2c4/)

[**N-Queens**](https://www.hackerearth.com/practice/basic-programming/recursion/recursion-and-backtracking/practice-problems/algorithm/n-queensrecursion-tutorial/)

[**Simran and stairs**](https://www.hackerearth.com/practice/basic-programming/recursion/recursion-and-backtracking/practice-problems/algorithm/simran-and-stairs/)

[**biggest forest**](https://www.hackerearth.com/practice/basic-programming/recursion/recursion-and-backtracking/practice-problems/algorithm/biggest-forest-700592dd/)

[**Converting**](https://www.hackerearth.com/practice/basic-programming/recursion/recursion-and-backtracking/practice-problems/algorithm/question-2-38-cf73c1b4/)

[**Divide Number**](https://www.hackerearth.com/practice/basic-programming/recursion/recursion-and-backtracking/practice-problems/algorithm/divide-number-a410603f/)

[**Encrypted Love 2.O**](https://www.hackerearth.com/practice/basic-programming/recursion/recursion-and-backtracking/practice-problems/algorithm/encrypted-love-2o/)

[**Number of divisors**](https://www.hackerearth.com/practice/basic-programming/recursion/recursion-and-backtracking/practice-problems/algorithm/k-excess-1-be669e5a/)

1. [Backtracking | Set 1 (The Knight’s tour problem)](https://www.geeksforgeeks.org/backtracking-set-1-the-knights-tour-problem/)
2. [Backtracking | Set 2 (Rat in a Maze)](https://www.geeksforgeeks.org/backttracking-set-2-rat-in-a-maze/)
3. [Backtracking | Set 3 (N Queen Problem)](https://www.geeksforgeeks.org/backtracking-set-3-n-queen-problem/)
4. [Backtracking | Set 4 (Subset Sum)](https://www.geeksforgeeks.org/backttracking-set-4-subset-sum/)
5. [Backtracking | Set 5 (m Coloring Problem)](https://www.geeksforgeeks.org/backttracking-set-5-m-coloring-problem/)
6. [Backtracking | Set 6 (Hamiltonian Cycle)](https://www.geeksforgeeks.org/backtracking-set-7-hamiltonian-cycle/)
7. [Backtracking | Set 7 (Sudoku)](https://www.geeksforgeeks.org/backtracking-set-7-suduku/)
8. [Backtracking | Set 8 (Solving Cryptarithmetic Puzzles)](https://www.geeksforgeeks.org/backtracking-set-8-solving-cryptarithmetic-puzzles/)
9. [Backtracking | Set 9 (Magnet Puzzle)](https://www.geeksforgeeks.org/backtracking-set-9-magnet-puzzle/)
10. [N Queen in O(n) space](https://www.geeksforgeeks.org/n-queen-in-on-space/)
11. [Boggle | Set 2 (Using Trie)](https://www.geeksforgeeks.org/boggle-set-2-using-trie/)
12. [Remove Invalid Parentheses](https://www.geeksforgeeks.org/remove-invalid-parentheses/)
13. [Prime numbers after prime P with sum S](https://www.geeksforgeeks.org/prime-numbers-after-prime-p-with-sum-s/)
14. [Rat in a Maze with multiple steps or jump allowed](https://www.geeksforgeeks.org/rat-in-a-maze-with-multiple-steps-jump-allowed/)
15. [A backtracking approach to generate n bit Gray Codes](https://www.geeksforgeeks.org/backtracking-approach-generate-n-bit-gray-codes/)
16. [C++ program for Solving Cryptarithmetic Puzzles](https://www.geeksforgeeks.org/c-code-article-backtracking-set-8-solving-cryptarithmetic-puzzles/)
17. [Write a program to print all permutations of a given string](https://www.geeksforgeeks.org/write-a-c-program-to-print-all-permutations-of-a-given-string/)
18. [Print all possible paths from top left to bottom right of a mXn matrix](https://www.geeksforgeeks.org/print-all-possible-paths-from-top-left-to-bottom-right-of-a-mxn-matrix/)

FOR MORE:

<https://leetcode.com/tag/recursion/>

<https://leetcode.com/tag/backtracking/>