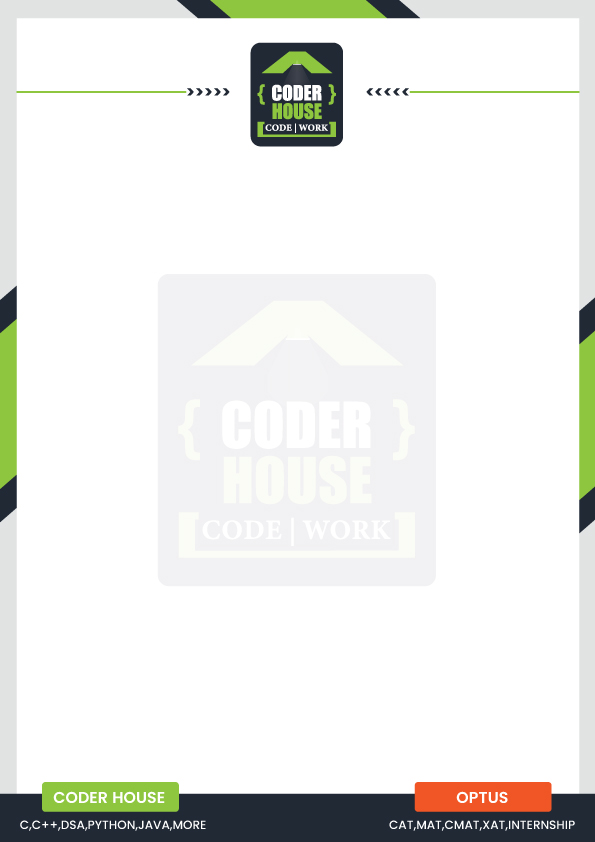


|  |  |
| --- | --- |
| Days | Questions |
| DAY 1 | **Find the maximum and minimum elements in an array.** |
| DAY 2 | **Reverse an array.** |
| DAY 3 | **Find the "Kth" smallest/largest element in an array.** |
| DAY 4 | **Move all negative elements to one side of the array.** |
| DAY 5 | **Sort an array of 0s, 1s, and 2s without using extra space.** |
| DAY 6 | **Find the union and intersection of two arrays.** |
| DAY 7 | **Rotate an array by "K" steps (cyclic rotation).** |
| DAY 8 | **Find the duplicate in an array of "N+1" integers.** |
| DAY 9 | **Find the missing number in an array of size "N".** |
| DAY 10 | **Check if an array is a palindrome.** |
| DAY 11 | **Merge two sorted arrays without extra space.** |
| DAY 12 | **Find the subarray with the maximum sum (Kadane’s Algorithm).** |
| DAY 13 | **Find the longest consecutive sequence in an array.** |
| DAY 14 | **Find the common elements in three sorted arrays.** |
| DAY 15 | **Find the majority element (element appearing more than N/2 times).** |



|  |  |
| --- | --- |
| Days | Questions |
| DAY 16 | **Rearrange the array in alternating positive and negative numbers.** |
| DAY 17 | **Find all subarrays with a given sum.** |
| DAY 18 | **Find the first repeating element in an array.** |
| DAY 19 | **Find the length of the largest subarray with contiguous elements.** |
| DAY 20 | **Rearrange the array to maximize the difference between consecutive elements.** |
| DAY 21 | **Find the minimum swaps required to bring all elements less than or equal to "K" together.** |
| DAY 22 | **Count inversions in an array (using merge sort).** |
| DAY 23 | **Find the "Kth" largest element in a stream.** |
| DAY 24 | **Find the maximum product subarray.** |
| DAY 25 | **Find the median of two sorted arrays of different sizes.** |
| DAY 26 | **Implement a program to merge overlapping intervals.** |
| DAY 27 | **Find the triplet that sums to a given value.** |
| DAY 28 | **Find the maximum sum of "K" consecutive elements in an array (sliding window).** |
| DAY 29 | **Find the minimum number of platforms required for trains (arrival and departure times).** |
| DAY 30 | **Find the minimum distance between two given elements in an array.** |