

HASHING

1. [Introduction](#)
2. [Index Mapping \(or Trivial Hashing\)](#)
3. [Separate Chaining for Collision Handling](#)
4. [Open Addressing for Collision Handling](#)
5. [Double Hashing](#)
6. [Load Factor and Rehashing](#) (isme ekk code hai ushe ache se pad lena... hashing ka exact code hai ye)

HASHING QUESTION

1. [Find whether an array is subset of another array](#)
2. [Given an array A\[\] and a number x, check for pair in A\[\] with sum as x](#) (company)
3. [Maximum distance between two occurrences of same element in array](#)
4. [Count maximum points on same line](#) (company)
5. [Check if a given array contains duplicate elements within k distance from each other](#)
6. [Find top k \(or most frequent\) numbers in a stream](#)
7. [First element occurring k times in an array](#)
8. [Pair with given sum and maximum shortest distance from end](#)
9. [k-th missing element in increasing sequence which is not present in a given sequence](#)
10. [Count items common to both the lists but with different prices](#)
11. [Count pairs whose products exist in array](#)
12. [Sort elements by frequency](#)
13. [Group words with same set of characters](#)
14. [Find k numbers with most occurrences in the given array](#)
15. [Find Itinerary from a given list of tickets](#) (company)
16. [Find number of Employees Under every Employee](#)
17. [Check if an array can be divided into pairs whose sum is divisible by k](#)
18. [Find four elements a, b, c and d in an array such that \$a+b = c+d\$](#)
19. [Find the largest subarray with 0 sum](#) (company)
20. [Longest Increasing consecutive subsequence](#)
21. [Longest subsequence such that difference between adjacents is one | Set 2](#)
22. [Longest Consecutive Subsequence](#)
23. [Count distinct elements in every window of size k](#)
24. [Design a data structure that supports insert, delete, search and getRandom in constant time](#)
25. [Length of the largest subarray with contiguous elements](#)
26. [Group Shifted String](#)
27. [Minimum insertions to form a palindrome with permutations allowed](#)

28. [Check for Palindrome after every character replacement Query](#)
29. [Maximum difference between first and last indexes of an element in array](#)
30. [Smallest subarray with k distinct numbers](#)
31. [Largest subarray with equal number of 0s and 1s \(company\)](#)
32. [Count Substrings with equal number of 0s, 1s and 2s](#)
33. [Print all triplets in sorted array that form AP](#)
34. [Find all triplets with zero sum](#)
35. [Palindrome Substring Queries](#)
36. [Find smallest range containing elements from k lists](#)
37. [Range Queries for Frequencies of array elements](#)
38. [Elements to be added so that all elements of a range are present in array](#)
39. [Subarrays with distinct elements](#)
40. [Count subarrays having total distinct elements same as original array](#)
41. [Count subarrays with same even and odd elements](#)
42. [Minimum number of distinct elements after removing m items](#)
43. [Distributing items when a person cannot take more than two items of same type](#)
44. [Maximum consecutive numbers present in an array](#)
45. [Maximum number of chocolates to be distributed equally among k students \(company\)](#)
46. [Find Sum of all unique sub-array sum for a given array.](#)