

Basic DSA Sheet

[c++](#)

[c++STL](#)

[Basic Programs](#)

ARRAY

- 1 [Two Sum](#)
- 2 [Remove Duplicates from Sorted Array](#)
- 3 [Remove Element](#)
- 4 [Search Insert Position](#)
- 5 [Maximum Subarray](#)
- 6 [Plus One](#)
- 7 [Merge Sorted Array](#)
- 8 [Best Time to Buy and Sell Stock](#)
- 9 [Two Sum II - Input array is sorted](#)
- 10 [Majority Element](#)

MATH

- 1 [Reverse Integer](#)
- 2 [Palindrome Number](#)
- 3 [Sort\(x\)](#)
- 4 [Excel Sheet Column Title](#)
- 5 [Excel Sheet Column Number](#)
- 6 [Factorial Trailing Zeroes](#)
- 7 [Happy Number](#)
- 8 [Count Primes](#)
- 9 [Power of Two](#)
- 10 [Add Digits](#)

LINKED LIST

- 1 [Add Two Numbers](#)
- 2 [Reverse Linked List](#)
- 3 [Convert Binary Number in a Linked List to Integer](#)
- 4 [Middle of the Linked List](#)
- 5 [Remove Linked List Elements](#)
- 6 [Delete Node in a Linked List](#)
- 7 [Intersection of Two Linked Lists](#)
- 8 [Remove Duplicates from Sorted List](#)
- 9 [Merge Two Sorted Lists](#)
- 10 [Linked List Cycle](#)

TWO POINTER

- 1 [Roman to Integer](#)
- 2 [Valid Palindrome](#)
- 3 [Move Zeroes](#)
- 4 [Reverse String](#)
- 5 [Squares of a Sorted Array](#)
- 6 [Long Pressed Name](#)
- 7 [Backspace String Compare](#)
- 8 [Intersection of Two Arrays](#)
- 9 [Intersection of Two Arrays II](#)
- 10 [Implement strStr\(\)](#)

MY LINKEDLN PROFILE

STRINGS

- 1 [Defanging an IP Address](#)
- 2 [Consecutive Characters](#)
- 3 [Robot Return to Origin](#)
- 4 [To Lower Case](#)
- 5 [Split a String in Balanced Strings](#)
- 6 [Maximum Number of Balloons](#)
- 7 [Length of Last Word](#)
- 8 [Valid Palindrome II](#)
- 9 [Goat Latin](#)
- 10 [Reverse Words in a String III](#)

BINARY SEARCH

- 1 [Search a 2D Matrix II](#)
- 2 [Pow\(x, n\)](#)
- 3 [Valid Perfect Square](#)
- 4 [Guess Number Higher or Lower](#)
- 5 [Is Subsequence](#)
- 6 [First Bad Version](#)
- 7 [Arranging Coins](#)
- 8 [Binary Search](#)
- 9 [Find Smallest Letter Greater Than Target](#)
- 10 [Peak Index in a Mountain Array](#)

STACK

- 1 [Remove Outermost Parentheses](#)
- 2 [Build an Array With Stack Operations](#)
- 3 [Backspace String Compare](#)
- 4 [Remove All Adjacent Duplicates In String](#)
- 5 [Baseball Game](#)
- 6 [Next Greater Element I](#)
- 7 [Implement Queue using Stacks](#)
- 8 [Implement Stack using Queues](#)
- 9 [Min Stack](#)
- 10 [Valid Parentheses](#)

HASH TABLE

- 1 [Jewels and Stones](#)
- 2 [N-Repeated Element in Size 2N Array](#)
- 3 [Find Words That Can Be Formed by Characters](#)
- 4 [Daily Temperatures](#)
- 5 [How Many Numbers Are Smaller Than the Current Number](#)
- 6 [Single Number](#)
- 7 [Contains Duplicate](#)
- 8 [Valid Anagram](#)
- 9 [Contains Duplicate II](#)
- 10 [Find the Difference](#)

TREE

- 1 [Range Sum of BST](#)
- 2 [Same Tree](#)
- 3 [Symmetric Tree](#)
- 4 [Maximum Depth of Binary Tree](#)
- 5 [Search in a Binary Search Tree](#)