

25 Most Asked DSA Questions In MAANG



1. Product of Array Except Self

Given an array nums of n integers, return an array output such that output[i] is equal to the product of all the elements of nums except nums[i].

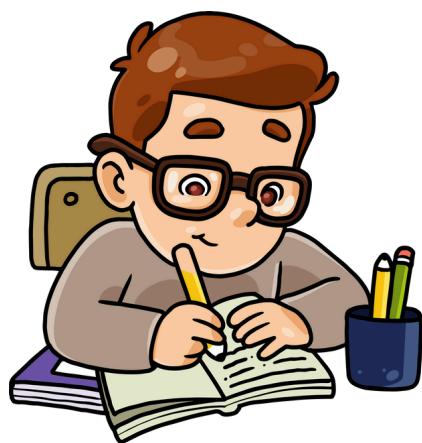
Example:

Input: [1, 2, 3, 4]

Output: [24, 12, 8, 6]

Explanation:

- $\text{output}[0] = 2 * 3 * 4 = 24$
- $\text{output}[1] = 1 * 3 * 4 = 12$
- $\text{output}[2] = 1 * 2 * 4 = 8$
- $\text{output}[3] = 1 * 2 * 3 = 6$



2 . 3Sum

Given an array nums of n integers, are there elements a, b, c in nums such that a + b + c = 0? Find all unique triplets in the array which gives the sum of zero.

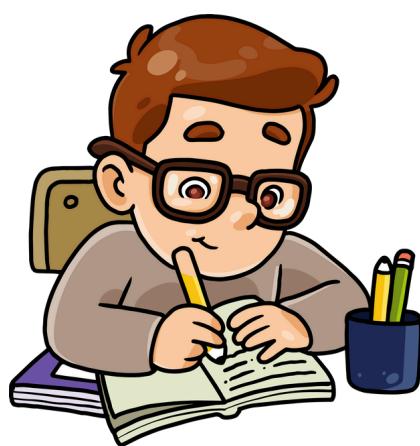
Notice that the solution set must not contain duplicate triplets.

Example:

Input: nums = [-1,0,1,2,-1,-4]

Output: [[-1,-1,2],[-1,0,1]]

Explanation: The solution set contains unique triplets [-1, 0, 1] and [-1, -1, 2].



3. Longest Palindromic Substring

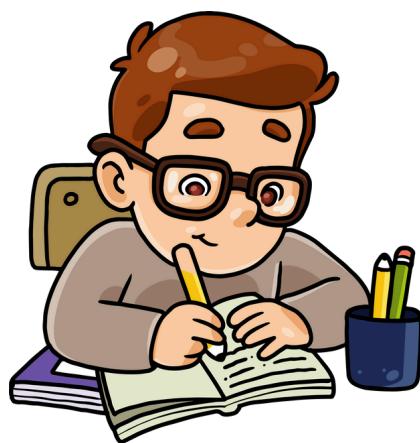
Given a string s, return the longest palindromic substring in s.

Example:

Input: s = "babad"

Output: "bab"

Note: "aba" is also a valid answer.



4 . Two Sum

Given an array of integers nums and an integer target, return indices of the two numbers such that they add up to target.

You may assume that each input would have exactly one solution, and you may not use the same element twice.

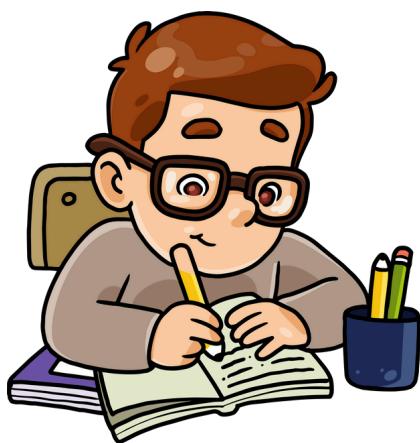
You can return the answer in any order.

Example:

Input: nums = [2,7,11,15], target = 9

Output: [0,1]

Explanation: Because nums[0] + nums[1] == 9, we return [0,1].



5 . Merge k Sorted Lists

You are given an array of k linked-lists lists, each linked-list is sorted in ascending order.

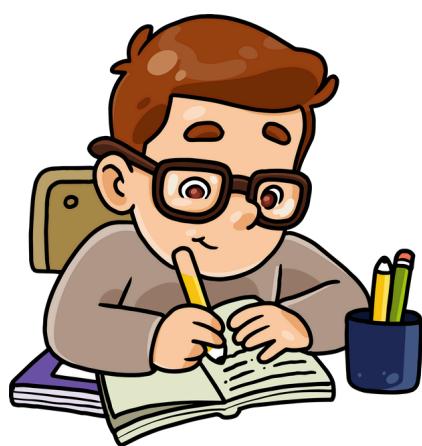
Merge all the linked-lists into one sorted linked-list and return it.

Example:

Input: lists = [[1,4,5],[1,3,4],[2,6]]

Output: [1,1,2,3,4,4,5,6]

Explanation: The linked-lists are merged into a single sorted linked-list in ascending order.



6 . Meeting Rooms

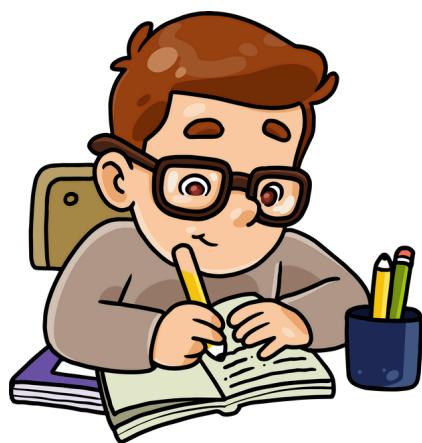
Given an array of meeting time intervals intervals where intervals[i] = [starti, endi], return true if you can attend all meetings, or false otherwise.

Example:

Input: intervals = [[0,30],[5,10],[15,20]]

Output: false

Explanation: The meeting time interval [0,30] is overlapping with [5,10], so it is impossible to attend all meetings without overlapping. Therefore, the function should return false.



7 . Reverse Linked Lists

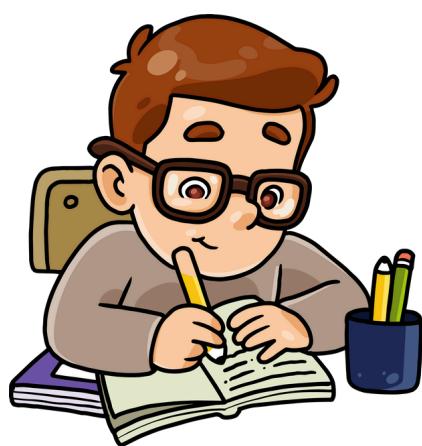
Reverse a singly linked list.

Example:

Input: 1 → 2 → 3 → 4 → 5 → NULL

Output: 5 → 4 → 3 → 2 → 1 → NULL

Explanation: The reversed linked list has its elements in reverse order compared to the original linked list.



8 . Longest Substring Without Repeating Characters

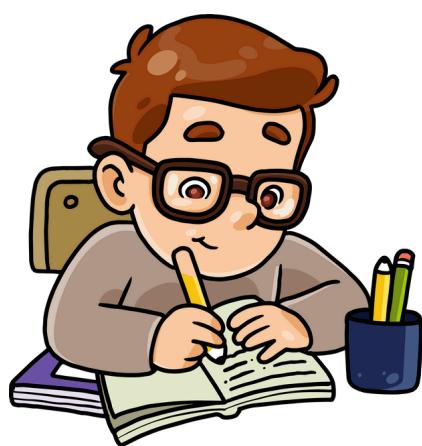
Given a string s, find the length of the longest substring without repeating characters.

Example:

Input: s = "abcabcbb"

Output: 3

Explanation: The longest substring without repeating characters is "abc", which has a length of 3.



9 . Merge k Sorted Lists

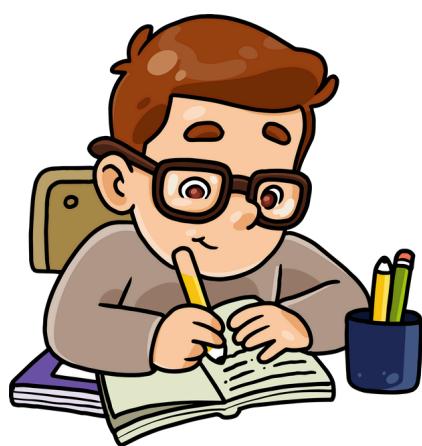
You're given an array of k linked lists, each linked list is sorted in ascending order. Merge all the linked lists into one sorted linked list and return it.

Example:

Input: lists = [[1,4,5],[1,3,4],[2,6]]

Output: [1,1,2,3,4,4,5,6]

Explanation: The linked lists are merged into a single sorted linked list in ascending order.



10 . Top k Frequent Elements

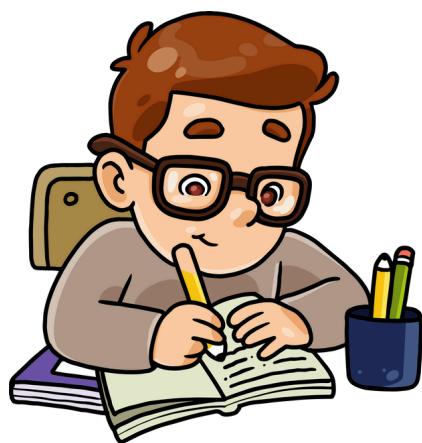
Given an integer array nums and an integer k, return the k most frequent elements. You may return the answer in any order.

Example:

Input: nums = [1,1,1,2,2,3], k = 2

Output: [1,2]

Explanation: Both 1 and 2 appear three times in the array, so the top 2 frequent elements are 1 and 2.



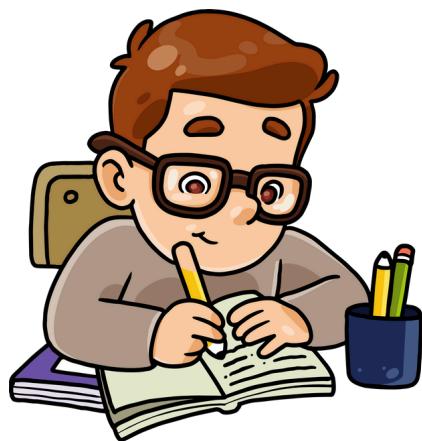
11. LRU Cache

Design and implement a data structure for Least Recently Used (LRU) cache with get and put operations.

Example:

```
LRUCache cache = new LRUCache(2); // capacity is 2
```

```
cache.put(1, 1);
cache.put(2, 2);
cache.get(1); // returns 1
cache.put(3, 3); // evicts key 2
cache.get(2); // returns -1
cache.put(4, 4); // evicts key 1
cache.get(1); // returns -1
cache.get(3); // returns 3
cache.get(4); // returns 4
```



12. Course Schedule

There are a total of numCourses courses you have to take, labeled from 0 to numCourses – 1. You are given an array prerequisites where prerequisites[i] = [ai, bi] indicates that you must take course bi first if you want to take course ai.

For example, the pair [0, 1], indicates that to take course 0 you have to first take course 1.

Return true if you can finish all courses. Otherwise, return false.

Example:

Input: numCourses = 2, prerequisites = [[1,0]]

Output: true

Explanation: There are a total of 2 courses to take. To take course 1 you should have finished course 0. So it is possible.

13 . Linked List Cycle

Given head, the head of a linked list, determine if the linked list has a cycle in it.

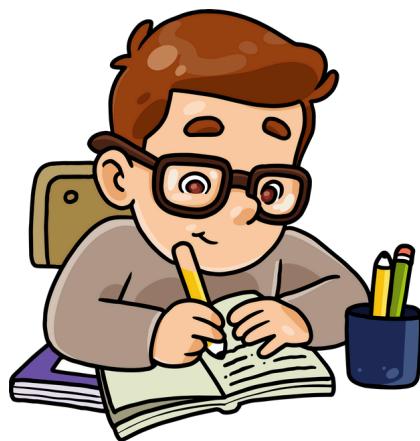
There is a cycle in a linked list if there is some node in the list that can be reached again by continuously following the next pointer. Return true if there is a cycle in the linked list. Otherwise, return false.

Example:

Input: head = [3,2,0,-4], pos = 1

Output: true

Explanation: There is a cycle in the linked list, where the tail connects to the 1st node (0-indexed), and the cycle's length is 3.



14 . Trapping Rainwater

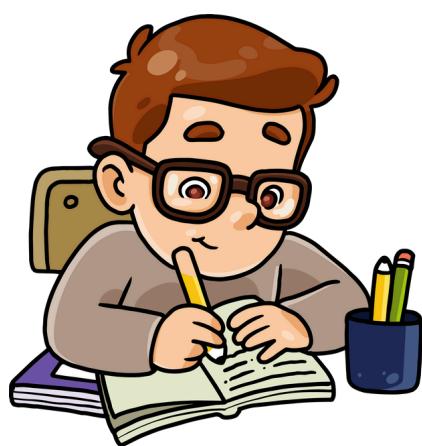
Given n non-negative integers representing an elevation map where the width of each bar is 1, compute how much water it can trap after raining.

Example:

Input: height = [0,1,0,2,1,0,1,3,2,1,2,1]

Output: 6

Explanation: The elevation map is represented by the array [0,1,0,2,1,0,1,3,2,1,2,1]. In this case, 6 units of rainwater (blue section) are being trapped.



15 . Median of Two Sorted Arrays

Given two sorted arrays `nums1` and `nums2` of size m and n respectively, return the median of the two sorted arrays.

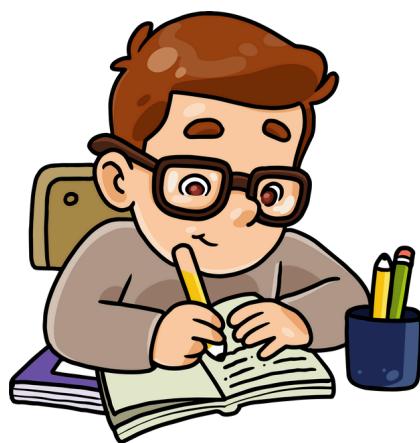
Follow up: The overall run time complexity should be $O(\log(m+n))$.

Example:

Input: `nums1 = [1, 3]`, `nums2 = [2]`

Output: 2.0

Explanation: The median is 2.0 since it is the average of 1 and 3.



16 . Best Time to buy and sell stock

You are given an array `prices` where `prices[i]` is the price of a given stock on the `i`th day. You want to maximize your profit by choosing a single day to buy one stock and choosing a different day in the future to sell that stock.

Return the maximum profit you can achieve from this transaction. If you cannot achieve any profit, return 0.

Example:

Input: `prices = [7,1,5,3,6,4]`

Output: 5

Explanation: Buy on day 2 (price = 1) and sell on day 3 (price = 5), profit = $5 - 1 = 4$.

Note: Buying on day 2 and selling on day 1 is not allowed because you must buy before you sell.

17 . Longest Common Prefix

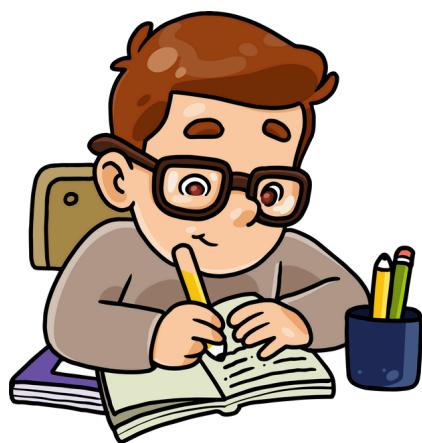
Write a function to find the longest common prefix string amongst an array of strings. If there is no common prefix, return an empty string "".

Example:

Input: ["flower", "flow", "flight"]

Output: "fl"

Explanation: The longest common prefix is "fl".



18 . Add two Numbers

You are given two non-empty linked lists representing two non-negative integers. The digits are stored in reverse order, and each of their nodes contains a single digit. Add the two numbers and return the sum as a linked list.

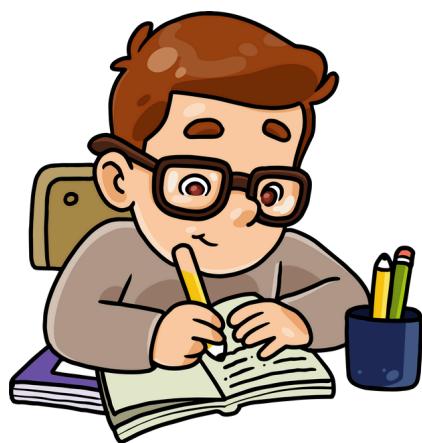
You may assume the two numbers do not contain any leading zero, except the number 0 itself.

Example:

Input: l1 = [2,4,3], l2 = [5,6,4]

Output: [7,0,8]

Explanation: $342 + 465 = 807$.



19 . Non-Overlapping Intervals

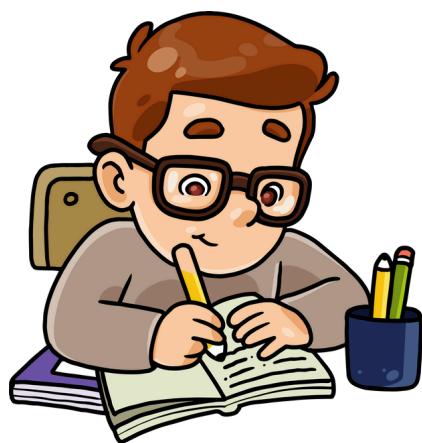
Given a collection of intervals, find the minimum number of intervals you need to remove to make the rest of the intervals non-overlapping.

Example:

Input: [[1,2],[2,3],[3,4],[1,3]]

Output: 1

Explanation: The minimum number of intervals to remove is 1, which can be achieved by removing the interval [1,3]. After removing this interval, the rest of the intervals [1,2], [2,3], and [3,4] are non-overlapping.



20 . Maximum Subarray

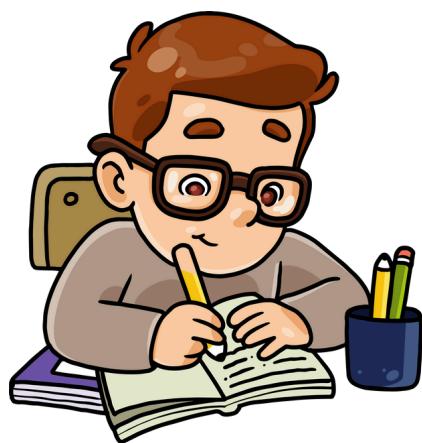
Given an integer array nums, find the contiguous subarray (containing at least one number) which has the largest sum and return its sum.

Example:

Input: nums = [-2,1,-3,4,-1,2,1,-5,4]

Output: 6

Explanation: The contiguous subarray [4,-1,2,1] has the largest sum = 6.



21 . Search in Rotated Sorted Array

Suppose an array of length n is sorted in ascending order and is rotated at some pivot unknown to you beforehand. You are given a target value to search. If found in the array return its index, otherwise return -1.

You may assume no duplicate exists in the array.

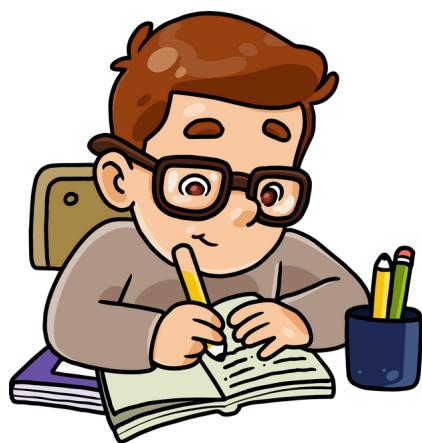
Your algorithm's runtime complexity must be in the order of $O(\log n)$.

Example:

Input: nums = [4,5,6,7,0,1,2], target = 0

Output: 4

Explanation: The index of 0 in the array is 4.



22 . Permutations

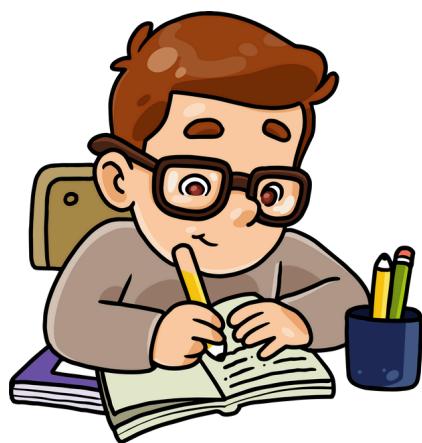
Given an array nums of distinct integers, return all possible permutations. You can return the answer in any order.

Example:

Input: nums = [1,2,3]

Output: [[1,2,3],[1,3,2],[2,1,3],[2,3,1],[3,1,2],[3,2,1]]

Explanation: These are all the possible permutations of the array [1,2,3].



23 . Word Search

Given a 2D board and a word, find if the word exists in the grid.

The word can be constructed from letters of sequentially adjacent cells, where "adjacent" cells are horizontally or vertically neighboring. The same letter cell may not be used more than once.

Example:

Board:

```
[  
  ['A','B','C','E'],  
  ['S','F','C','S'],  
  ['A','D','E','E']  
]
```

Given word = "ABCED", return true.

Given word = "SEE", return true.

Given word = "ABCB", return false.

Explanation:

- The first example matches the path "ABCED" in the board.
- The second example matches the path "SEE" in the board.
- The third example, although "ABC" exists in the board, it is not contiguous with "B".

24 . Number of Islands

Given an $m \times n$ 2D binary grid grid which represents a map of '1's (land) and '0's (water), return the number of islands.

An island is surrounded by water and is formed by connecting adjacent lands horizontally or vertically. You may assume all four edges of the grid are all surrounded by water.

Example:

Input: grid = [
 ["1", "1", "1", "1", "0"],
 ["1", "1", "0", "1", "0"],
 ["1", "1", "0", "0", "0"],
 ["0", "0", "0", "0", "0"]
]

Output: 1

Explanation: The example represents one island formed by '1's.

25 . Valid Parentheses

Given a string s containing just the characters '(', ')', '{', '}', '[' and ']', determine if the input string is valid.

An input string is valid if:

- 1. Open brackets must be closed by the same type of brackets.**
- 2. Open brackets must be closed in the correct order.**

Example:

Input: s = "()[]{}"

Output: true

Explanation: The string is valid because all parentheses are closed correctly.

