16. 3Sum Closest (Medium)

Given an array S of n integers, find three integers in S such that the sum is closest to a given number, target. Return the sum of the three integers. You may assume that each input would have exactly one solution.

```
    For example, given array S = {-1 2 1 -4}, and target = 1.

    The sum that is closest to the target is 2. (-1 + 2 + 1 = 2).
```

Companies

Bloomberg

Topics

Array   Two Pointers

259. 3Sum Smaller (Medium)

Given an array of n integers nums and a target, find the number of index triplets `i`, `j`, `k` with `0 <= i < j < k < n` that satisfy the condition `nums[i] + nums[j] + nums[k] < target`.

For example, given nums = `[-2, 0, 1, 3]`, and target = 2.

Return 2. Because there are two triplets which sums are less than 2:

```
[-2, 0, 1]
[-2, 0, 3]
```

Follow up:
Could you solve it in O(n^2) runtime?

Companies

Google

Topics

Array   Two Pointers

15. 3Sum (Medium)

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

Note: The solution set must not contain duplicate triplets.

```
For example, given array S = [-1, 0, 1, 2, -1, -4],

A solution set is:
[
  [-1, 0, 1],
  [-1, -1, 2]
]
```

Companies

Facebook   Microsoft   Amazon   Bloomberg   Adobe   Works Applications

Topics

Array   Two Pointers

18. 4Sum (Medium)

Given an array S of n integers, are there elements a, b, c, and d in S such that a + b + c + d = target? Find all unique quadruplets in the array which gives the sum of target.

Note: The solution set must not contain duplicate quadruplets.

```
For example, given array S = [1, 0, -1, 0, -2, 2], and target = 0.

A solution set is:
[
  [-1,  0, 0, 1],
  [-2, -1, 1, 2],
  [-2,  0, 0, 2]
]
```

Companies

Topics

Array   Hash Table   Two Pointers

211. Add and Search Word - Data structure design (Medium)

Design a data structure that supports the following two operations:

```
void addWord(word)
bool search(word)
```

search(word) can search a literal word or a regular expression string containing only letters `a-z` or `..` A `.` means it can represent any one letter.

For example:

```
addWord("bad")
addWord("dad")
addWord("mad")
search("pad") -> false
search("bad") -> true
search(".ad") -> true
search("b..") -> true
```

Note:
You may assume that all words are consist of lowercase letters `a-z`.

You should be familiar with how a Trie works. If not, please work on this problem: Implement Trie (Prefix Tree) first.

Companies

(Facebook)

Topics

(Backtracking) (Design) (Trie)

67. Add Binary (Easy)

Given two binary strings, return their sum (also a binary string).

For example,
a = "11"
b = "1"
Return "100".

Companies

Facebook

Topics

Math String

258. Add Digits (Easy)

Given a non-negative integer `num`, repeatedly add all its digits until the result has only one digit.

For example:

Given `num = 38`, the process is like: `3 + 8 = 11`, `1 + 1 = 2`. Since `2` has only one digit, return it.

Follow up:
Could you do it without any loop/recursion in O(1) runtime?

Companies

Microsoft   Adobe

Topics

Math

415. Add Strings (Easy)

Given two non-negative integers `num1` and `num2` represented as string, return the sum of `num1` and `num2`.

Note:

1. The length of both `num1` and `num2` is < 5100.
2. Both `num1` and `num2` contains only digits `0-9`.
3. Both `num1` and `num2` does not contain any leading zero.
4. You must not use any built-in BigInteger library   or convert the inputs to integer directly.

Companies

Google   Airbnb

Topics

Math

2. Add Two Numbers (Medium)

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 contain a single digit. Add the two numbers and return it as a linked list.

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

```
Input: (2 -> 4 -> 3) + (5 -> 6 -> 4)
Output: 7 -> 0 -> 8
```

Companies

Microsoft   Amazon   Bloomberg   Airbnb   Adobe

Topics

Linked List   Math

306. Additive Number (Medium)

Additive number is a string whose digits can form additive sequence.

A valid additive sequence should contain at least three numbers. Except for the first two numbers, each subsequent number in the sequence must be the sum of the preceding two.

For example:
"112358" is an additive number because the digits can form an additive sequence: 1, 1, 2, 3, 5, 8.

```
1 + 1 = 2, 1 + 2 = 3, 2 + 3 = 5, 3 + 5 = 8
```

"199100199" is also an additive number, the additive sequence is: 1, 99, 100, 199.

```
1 + 99 = 100, 99 + 100 = 199
```

Note: Numbers in the additive sequence cannot have leading zeros, so sequence 1, 2, 03 or 1, 02, 3 is invalid.

Given a string containing only digits '0'-'9', write a function to determine if it's an additive number.

Follow up:
How would you handle overflow for very large input integers?

Companies

Epic Systems

Topics

269. Alien Dictionary (Hard)

There is a new alien language which uses the latin alphabet. However, the order among letters are unknown to you. You receive a list of non-empty words from the dictionary, where words are sorted lexicographically by the rules of this new language . Derive the order of letters in this language.

Example 1:
Given the following words in dictionary,

```
[
  "wrt",
  "wrf",
  "er",
  "ett",
  "rftt"
]
```

The correct order is: `"wertf"` .

Example 2:
Given the following words in dictionary,

```
[
  "z",
  "x"
]
```

The correct order is: `"zx"` .

Example 3:
Given the following words in dictionary,

```
[
  "z",
  "x",
  "z"
]
```

The order is invalid, so return `""`.

Note:

1. You may assume all letters are in lowercase.
2. You may assume that if a is a prefix of b, then a must appear before b in the given dictionary.
3. If the order is invalid, return an empty string.
4. There may be multiple valid order of letters, return any one of them is fine.

Companies

Google Facebook Twitter Snapchat Airbnb Pocket Gems

Topics

Graph Topological Sort

351. Android Unlock Patterns (Medium)

Given an Android 3x3 key lock screen and two integers m and n, where 1 ≤ m ≤ n ≤ 9, count the total number of unlock patterns of the Android lock screen, which consist of minimum of m keys and maximum n keys.

Rules for a valid pattern:

1. Each pattern must connect at least m keys and at most n keys.
2. All the keys must be distinct.
3. If the line connecting two consecutive keys in the pattern passes through any other keys, the other keys must have previously selected in the pattern. No jumps through non selected key is allowed.
4. The order of keys used matters.
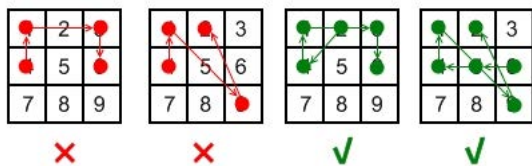
Explanation:

```
| 1 | 2 | 3 |
| 4 | 5 | 6 |
| 7 | 8 | 9 |
```

Invalid move: 4 - 1 - 3 - 6
Line 1 - 3 passes through key 2 which had not been selected in the pattern.

Invalid move: 4 - 1 - 9 - 2
Line 1 - 9 passes through key 5 which had not been selected in the pattern.

Valid move: 2 - 4 - 1 - 3 - 6
Line 1 - 3 is valid because it passes through key 2, which had been selected in the pattern

Valid move: 6 - 5 - 4 - 1 - 9 - 2

Line 1 - 9 is valid because it passes through key 5, which had been selected in the pattern.

Example:
Given m = 1, n = 1, return 9.

Companies

Google

Topics

Dynamic Programming Backtracking

413. Arithmetic Slices (Medium)

A sequence of number is called arithmetic if it consists of at least three elements and if the difference between any two consecutive elements is the same.

For example, these are arithmetic sequence:

```
1, 3, 5, 7, 9
7, 7, 7, 7
3, -1, -5, -9
```

The following sequence is not arithmetic.

```
1, 1, 2, 5, 7
```

A zero-indexed array A consisting of N numbers is given. A slice of that array is any pair of integers (P, Q) such that 0 <= P < Q < N.

A slice (P, Q) of array A is called arithmetic if the sequence:
A[P], A[p + 1], ..., A[Q - 1], A[Q] is arithmetic. In particular, this means that P + 1 < Q.

The function should return the number of arithmetic slices in the array A.

Example:

```
A = [1, 2, 3, 4]

return: 3, for 3 arithmetic slices in A: [1, 2, 3], [2, 3, 4] and
 [1, 2, 3, 4] itself.
```

Companies

Baidu   Aetion

Topics

110. Balanced Binary Tree (Easy)

Given a binary tree, determine if it is height-balanced.

For this problem, a height-balanced binary tree is defined as a binary tree in which the depth of the two subtrees of every node never differ by more than 1.

Companies

Bloomberg

Topics

Tree   Depth-first Search

227. Basic Calculator II (Medium)

Implement a basic calculator to evaluate a simple expression string.

The expression string contains only non-negative integers, `+`, `-`, `*`, `/` operators and empty spaces . The integer division should truncate toward zero.

You may assume that the given expression is always valid.

Some examples:

```
"3+2*2" = 7
" 3/2 " = 1
" 3+5 / 2 " = 5
```

Note: Do not use the `eval` built-in library function.

Companies

Airbnb

Topics

String

224. Basic Calculator (Hard)

Implement a basic calculator to evaluate a simple expression string.

The expression string may contain open (and closing parentheses), the plus + or minus sign - , non-negative integers and empty spaces .

You may assume that the given expression is always valid.

Some examples:

```
"1 + 1" = 2
" 2-1 + 2 " = 3
"(1+(4+5+2)-3)+(6+8)" = 23
```

Note: Do not use the `eval` built-in library function.

Companies

(Google)

Topics

(Math) (Stack)

296. Best Meeting Point (Hard)

A group of two or more people wants to meet and minimize the total travel distance. You are given a 2D grid of values 0 or 1, where each 1 marks the home of someone in the group. The distance is calculated using Manhattan Distance (https://en.wikipedia.org/wiki/Taxicab_geometry), where distance(p1, p2) = `|p2.x - p1.x| + |p2.y - p1.y|`.

For example, given three people living at `(0,0)`, `(0,4)`, and `(2,2)`:

```
1 - 0 - 0 - 0 - 1
|   |   |   |   |
0 - 0 - 0 - 0 - 0
|   |   |   |   |
0 - 0 - 1 - 0 - 0
```

The point `(0,2)` is an ideal meeting point, as the total travel distance of 2+2+2=6 is minimal. So return 6.

Companies

Twitter

Topics

Math   Sort

122. Best Time to Buy and Sell Stock II (Easy)

Say you have an array for which the ith element is the price of a given stock on day i.

Design an algorithm to find the maximum profit. You may complete as many transactions as you like (ie, buy one and sell one share of the stock multiple times). However, you may not engage in multiple transactions at the same time (ie, you must sell the stock before you buy again).

Companies

Bloomberg

Topics

Array   Greedy

123. Best Time to Buy and Sell Stock III (Hard)

Say you have an array for which the ith element is the price of a given stock on day i.

Design an algorithm to find the maximum profit. You may complete at most two transactions.

Note:
You may not engage in multiple transactions at the same time (ie, you must sell the stock before you buy again).

Companies

Topics

Array · Dynamic Programming

188. Best Time to Buy and Sell Stock IV (Hard)

Say you have an array for which the ith element is the price of a given stock on day i.

Design an algorithm to find the maximum profit. You may complete at most k transactions.

Note:
You may not engage in multiple transactions at the same time (ie, you must sell the stock before you buy again).

Companies

Topics

Dynamic Programming

309. Best Time to Buy and Sell Stock with Cooldown (Medium)

Say you have an array for which the ith element is the price of a given stock on day i.

Design an algorithm to find the maximum profit. You may complete as many transactions as you like (ie, buy one and sell one share of the stock multiple times) with the following restrictions:

- You may not engage in multiple transactions at the same time (ie, you must sell the stock before you buy again).
- After you sell your stock, you cannot buy stock on next day. (ie, cooldown 1 day)

Example:

```
prices = [1, 2, 3, 0, 2]
maxProfit = 3
transactions = [buy, sell, cooldown, buy, sell]
```

Companies

Google

Topics

Dynamic Programming

121. Best Time to Buy and Sell Stock (Easy)

Say you have an array for which the ith element is the price of a given stock on day i.

If you were only permitted to complete at most one transaction (ie, buy one and sell one share of the stock), design an algorithm to find the maximum profit.

Example 1:

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

max. difference = 6-1 = 5 (not 7-1 = 6, as selling price needs to be
larger than buying price)
```

Example 2:

```
Input: [7, 6, 4, 3, 1]
Output: 0

In this case, no transaction is done, i.e. max profit = 0.
```

Companies

Facebook   Microsoft   Amazon   Bloomberg   Uber

Topics

Array   Dynamic Programming

173. Binary Search Tree Iterator (Medium)

Implement an iterator over a binary search tree (BST). Your iterator will be initialized with the root node of a BST.

Calling `next()` will return the next smallest number in the BST.

Note: `next()` and `hasNext()` should run in average O(1) time and uses O(h) memory, where h is the height of the tree.

Companies

Google Facebook Microsoft LinkedIn

Topics

Stack Tree Design

94. Binary Tree Inorder Traversal (Medium)

Given a binary tree, return the inorder traversal of its nodes' values.

For example:
Given binary tree `[1,null,2,3]`,

```
   1
    \
     2
    /
   3
```

return `[1,3,2]`.

Note: Recursive solution is trivial, could you do it iteratively?

Companies

Microsoft

Topics

Hash Table   Stack   Tree

107. Binary Tree Level Order Traversal II (Easy)

Given a binary tree, return the bottom-up level order traversal of its nodes' values. (ie, from left to right, level by level from leaf to root).

For example:
Given binary tree `[3,9,20,null,null,15,7]`,

```
   3
  / \
 9  20
   /  \
  15   7
```

return its bottom-up level order traversal as:

```
[
  [15,7],
  [9,20],
  [3]
]
```

Companies

Topics

Tree   Breadth-first Search

102. Binary Tree Level Order Traversal (Medium)

Given a binary tree, return the level order traversal of its nodes' values. (ie, from left to right, level by level).

For example:
Given binary tree `[3,9,20,null,null,15,7]`,

```
    3
   / \
  9   20
     /  \
    15   7
```

return its level order traversal as:

```
[
  [3],
  [9,20],
  [15,7]
]
```

Companies

Facebook   Microsoft   Amazon   Bloomberg   LinkedIn   Apple

Topics

Tree   Breadth-first Search

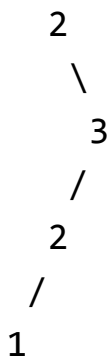## 298. Binary Tree Longest Consecutive Sequence (Medium)

Given a binary tree, find the length of the longest consecutive sequence path.

The path refers to any sequence of nodes from some starting node to any node in the tree along the parent-child connections. The longest consecutive path need to be from parent to child (cannot be the reverse).

For example,

```
   1
    \
     3
    / \
   2   4
        \
         5
```
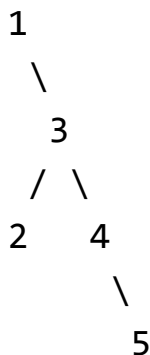
Longest consecutive sequence path is `3-4-5`, so return `3`.

```
     2
      \
       3
      /
     2
    /
   1
```

Longest consecutive sequence path is `2-3`, not `3-2-1`, so return `2`.
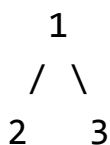
Companies

(Google)

Topics

(Tree)

124. Binary Tree Maximum Path Sum (Hard)

Given a binary tree, find the maximum path sum.

For this problem, a path is defined as any sequence of nodes from some starting node to any node in the tree along the parent-child connections. The path must contain at least one node   and does not need to go through the root.

For example:
Given the below binary tree,

```
    1
   / \
  2   3
```

Return 6 .
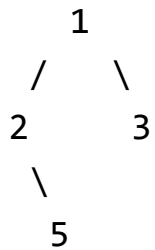
Companies

Microsoft   Baidu

Topics

Tree   Depth-first Search

257. Binary Tree Paths (Easy)

Given a binary tree, return all root-to-leaf paths.

For example, given the following binary tree:

```
    1
  /   \
 2     3
  \
   5
```

All root-to-leaf paths are:

```
["1->2->5", "1->3"]
```

Companies

Google Facebook Apple

Topics

Tree Depth-first Search

145. Binary Tree Postorder Traversal (Hard)

Given a binary tree, return the postorder traversal of its nodes' values.

For example:
Given binary tree `{1,#,2,3}`,

```
   1
    \
     2
    /
   3
```

return `[3,2,1]`.

Note: Recursive solution is trivial, could you do it iteratively?

Companies

Topics

Stack   Tree

144. Binary Tree Preorder Traversal (Medium)

Given a binary tree, return the preorder traversal of its nodes' values.

For example:
Given binary tree `[1,null,2,3]`,

```
   1
    \
     2
    /
   3
```

return `[1,2,3]`.

Note: Recursive solution is trivial, could you do it iteratively?
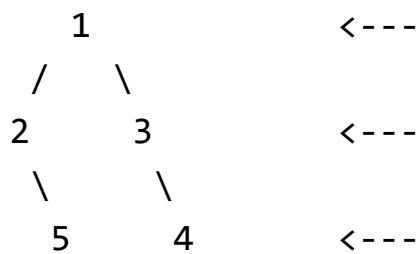
Companies

Topics

Stack   Tree

199. Binary Tree Right Side View (Medium)

Given a binary tree, imagine yourself standing on the right side of it, return the values of the nodes you can see ordered from top to bottom.

For example:
Given the following binary tree,

```
   1                <---
  / \
 2   3              <---
  \   \
   5   4            <---
```

You should return `[1, 3, 4]`.

Companies

Amazon

Topics

Tree   Depth-first Search   Breadth-first Search

156. Binary Tree Upside Down (Medium)

Given a binary tree where all the right nodes are either leaf nodes with a sibling (a left node that shares the same parent node) or empty, flip it upside down and turn it into a tree where the original right nodes turned into left leaf nodes. Return the new root.

For example:
Given a binary tree `{1,2,3,4,5}`,

```
    1
   / \
  2   3
 / \
4   5
```
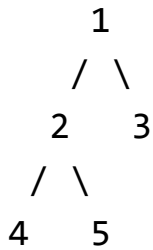
return the root of the binary tree `[4,5,2,#,#,3,1]`.

```
    4
   / \
  5   2
     / \
    3   1
```
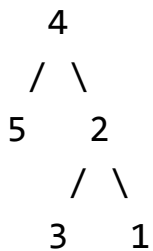
OJ's Binary T ree Serialization:

The serialization of a binary tree follows a level order traversal, where '#' signifies a path terminator where no node exists below.

Here's an example:

```
    1
   / \
  2   3
     /
    4
     \
      5
```
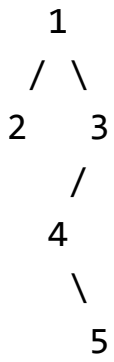
The above binary tree is serialized as `"{1,2,3,#,#,4,#,#,5}"`.

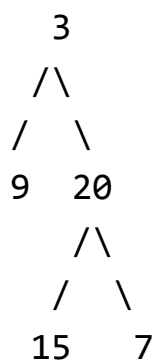## Companies

LinkedIn

Topics

Tree

314. Binary Tree Vertical Order Traversal (Medium)

Given a binary tree, return the vertical order traversal of its nodes' values. (ie, from top to bottom, column by column).

If two nodes are in the same row and column, the order should be from left to right .

Examples:

1. Given binary tree `[3,9,20,null,null,15,7]`,

```
    3
   /\
  /  \
  9   20
      /\
     /  \
    15   7
```

return its vertical order traversal as:

```
[
   [9],
   [3,15],
   [20],
   [7]
]
```

2. Given binary tree `[3,9,8,4,0,1,7]`,

```
      3
     /\
    /  \
    9   8
   /\  /\
  /  \/  \
  4  01   7
```
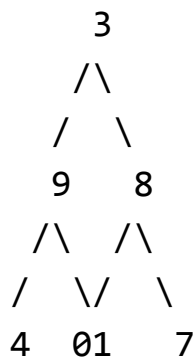
return its vertical order traversal as:

```
[
   [4],
   [9],
   [3,0,1],
   [8],
   [7]
]
```

3. Given binary tree `[3,9,8,4,0,1,7,null,null,null,2,5]` (0's right child is 2 and 1's left child is 5),

```
      3
     /\
    /  \
   9    8
  /\   /\
 /  \/  \
4   01   7
    /\
   /  \
  5    2
```
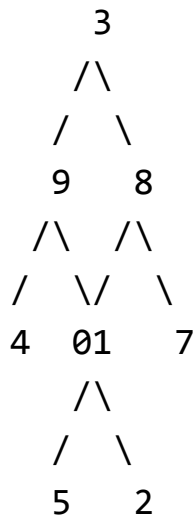
return its vertical order traversal as:

```
[
   [4],
   [9,5],
   [3,0,1],
   [8,2],
   [7]
]
```

Companies

Google Facebook Snapchat

Topics

Hash Table

103. Binary Tree Zigzag Level Order Traversal (Medium)

Given a binary tree, return the zigzag level order traversal of its nodes' values. (ie, from left to right, then right to left for the next level and alternate between).

For example:
Given binary tree `[3,9,20,null,null,15,7]`,

```
    3
   / \
  9  20
    /  \
   15   7
```

return its zigzag level order traversal as:

```
[
  [3],
  [20,9],
  [15,7]
]
```

Companies

Microsoft   Bloomberg   LinkedIn

Topics

Stack   Tree   Breadth-first Search

401. Binary Watch (Easy)

A binary watch has 4 LEDs on the top which represent the hours (0-11), and the 6 LEDs on the bottom represent the minutes (0-59).

Each LED represents a zero or one, with the least significant bit on the right.

For example, the above binary watch reads "3:25".

Given a non-negative integer n which represents the number of LEDs that are currently on, return all possible times the watch could represent.

Example:

```
Input: n = 1
Return: ["1:00", "2:00", "4:00", "8:00", "0:01", "0:02", "0:04", "0:
08", "0:16", "0:32"]
```

Note:

- The order of output does not matter.
- The hour must not contain a leading zero, for example "01:00" is not valid, it should be "1:00".
- The minute must be consist of two digits and may contain a leading zero, for example "10:2" is not valid, it should be "10:02".

Companies

Google

Topics

Backtracking   Bit Manipulation

201. Bitwise AND of Numbers Range (Medium)

Given a range [m, n] where 0 <= m <= n <= 2147483647, return the bitwise AND of all numbers in this range, inclusive.

For example, given the range [5, 7], you should return 4.

Companies

Topics

Bit Manipulation

361. Bomb Enemy (Medium)

Given a 2D grid, each cell is either a wall `'W'`, an enemy `'E'` or empty `'0'` (the number zero), return the maximum enemies you can kill using one bomb.
The bomb kills all the enemies in the same row and column from the planted point until it hits the wall since the wall is too strong to be destroyed.
Note that you can only put the bomb at an empty cell.

Example:

```
For the given grid

0 E 0 0
E 0 W E
0 E 0 0

return 3. (Placing a bomb at (1,1) kills 3 enemies)
```

Companies

Google

Topics

Dynamic Programming

319. Bulb Switcher (Medium)

There are n bulbs that are initially off. You first turn on all the bulbs. Then, you turn off every second bulb. On the third round, you toggle every third bulb (turning on if it's off or turning off if it's on). For the ith round, you toggle every i bulb. For the nth round, you only toggle the last bulb. Find how many bulbs are on after n rounds.

Example:

```
Given n = 3.

At first, the three bulbs are [off, off, off].
After first round, the three bulbs are [on, on, on].
After second round, the three bulbs are [on, off, on].
After third round, the three bulbs are [on, off, off].

So you should return 1, because there is only one bulb is on.
```

Companies

Topics

Math   Brainteaser

299. Bulls and Cows (Medium)

You are playing the following Bulls and Cows (https://en.wikipedia.org/wiki/Bulls_and_Cows) game with your friend: You write down a number and ask your friend to guess what the number is. Each time your friend makes a guess, you provide a hint that indicates how many digits in said guess match your secret number exactly in both digit and position (called "bulls") and how many digits match the secret number but locate in the wrong position (called "cows"). Your friend will use successive guesses and hints to eventually derive the secret number.

For example:

```
Secret number:  "1807"
Friend's guess: "7810"
```

Hint: `1` bull and `3` cows. (The bull is `8`, the cows are `0`, `1` and `7`.)

Write a function to return a hint according to the secret number and friend's guess, use `A` to indicate the bulls and `B` to indicate the cows. In the above example, your function should return `"1A3B"`.

Please note that both secret number and friend's guess may contain duplicate digits, for example:

```
Secret number:  "1123"
Friend's guess: "0111"
```

In this case, the 1st `1` in friend's guess is a bull, the 2nd or 3rd `1` is a cow, and your function should return `"1A1B"`.

You may assume that the secret number and your friend's guess only contain digits, and their lengths are always equal.

Companies

Topics

Hash Table

312. Burst Balloons (Hard)

Given `n` balloons, indexed from `0` to `n-1`. Each balloon is painted with a number on it represented by array `nums`. You are asked to burst all the balloons. If the you burst balloon `i` you will get `nums[left] * nums[i] * nums[right]` coins. Here `left` and `right` are adjacent indices of `i`. After the burst, the `left` and `right` then becomes adjacent.

Find the maximum coins you can collect by bursting the balloons wisely.

Note:
(1) You may imagine `nums[-1]` = `nums[n]` = `1`. They are not real therefore you can not burst them.
(2) 0 ≤ n ≤ 500, 0 ≤ `nums[i]` ≤ 100

Example:

Given `[3, 1, 5, 8]`

Return `167`

```
nums = [3,1,5,8] --> [3,5,8] -->   [3,8]   -->  [8]  --> []
coins =   3*1*5      +  3*5*8    +  1*3*8       + 1*8*1   = 167
```

Companies

Google   Snapchat

Topics

Divide and Conquer   Dynamic Programming

135. Candy (Hard)

There are N children standing in a line. Each child is assigned a rating value.

You are giving candies to these children subjected to the following requirements:

- Each child must have at least one candy.
- Children with a higher rating get more candies than their neighbors.

What is the minimum candies you must give?

Companies

Topics

Greedy

70. Climbing Stairs (Easy)

You are climbing a stair case. It takes n steps to reach to the top.

Each time you can either climb 1 or 2 steps. In how many distinct ways can you climb to the top?

Note: Given n will be a positive integer.

Example 1:

```
Input: 2
Output:  2
Explanation:  There are two ways to climb to the top.


1. 1 step + 1 step
2. 2 steps
```

Example 2:

```
Input: 3
Output:  3
Explanation:  There are three ways to climb to the top.


1. 1 step + 1 step + 1 step
2. 1 step + 2 steps
3. 2 steps + 1 step
```

Companies

Apple   Adobe

Topics

Dynamic Programming

133. Clone Graph (Medium)

Clone an undirected graph. Each node in the graph contains a `label` and a list of its `neighbors`.

OJ's undirected graph serialization:

Nodes are labeled uniquely.

We use `#` as a separator for each node, and `,` as a separator for node label and each neighbor of the node.

As an example, consider the serialized graph `{0,1,2#1,2#2,2}`.

The graph has a total of three nodes, and therefore contains three parts as separated by `#`.

1. First node is labeled as `0`. Connect node `0` to both nodes `1` and `2`.
2. Second node is labeled as `1`. Connect node `1` to node `2`.
3. Third node is labeled as `2`. Connect node `2` to node `2` (itself), thus forming a self-cycle.

Visually, the graph looks like the following:

```
     1
    / \
   /   \
  0 --- 2
       / \
       \_/
```
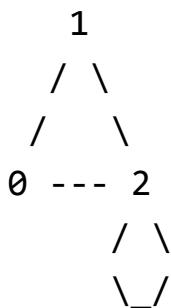
Companies

Google   Facebook   Uber   Pocket Gems

Topics

Depth-first Search   Breadth-first Search   Graph

272. Closest Binary Search Tree Value II (Hard)

Given a non-empty binary search tree and a target value, find k values in the BST that are closest to the target.

Note:

- Given target value is a floating point.
- You may assume k is always valid, that is: k ≤ total nodes.
- You are guaranteed to have only one unique set of k values in the BST that are closest to the target.

Follow up:
Assume that the BST is balanced, could you solve it in less than O(n) runtime (where n = total nodes)?

Companies

Google

Topics

Stack   Tree

270. Closest Binary Search Tree Value (Easy)

Given a non-empty binary search tree and a target value, find the value in the BST that is closest to the target.

Note:

- Given target value is a floating point.
- You are guaranteed to have only one unique value in the BST that is closest to the target.

Companies

Google Microsoft Snapchat

Topics

Binary Search Tree

322. Coin Change (Medium)

You are given coins of different denominations and a total amount of money amount. Write a function to compute the fewest number of coins that you need to make up that amount. If that amount of money cannot be made up by any combination of the coins, return `-1`.

Example 1:
coins = `[1, 2, 5]`, amount = `11`
return `3` (11 = 5 + 5 + 1)

Example 2:
coins = `[2]`, amount = `3`
return `-1`.

Note:
You may assume that you have an infinite number of each kind of coin.

Companies

Topics

(Dynamic Programming)

40. Combination Sum II (Medium)

Given a collection of candidate numbers (C) and a target number (T), find all unique combinations in C where the candidate numbers sums to T.

Each number in C may only be used once in the combination.

Note:

- All numbers (including target) will be positive integers.
- The solution set must not contain duplicate combinations.

For example, given candidate set `[10, 1, 2, 7, 6, 1, 5]` and target `8`,
A solution set is:

```
[
  [1, 7],
  [1, 2, 5],
  [2, 6],
  [1, 1, 6]
]
```

Companies

Snapchat

Topics

Array   Backtracking

216. Combination Sum III (Medium)

Find all possible combinations of k numbers that add up to a number n, given that only numbers from 1 to 9 can be used and each combination should be a unique set of numbers.

Example 1:

Input: k = 3, n = 7

Output:

```
[[1,2,4]]
```

Example 2:

Input: k = 3, n = 9

Output:

```
[[1,2,6], [1,3,5], [2,3,4]]
```

Companies

Topics

Array   Backtracking

377. Combination Sum IV (Medium)

Given an integer array with all positive numbers and no duplicates, find the number of possible combinations that add up to a positive integer target.

Example:

```
nums = [1, 2, 3]
target = 4

The possible combination ways are:
(1, 1, 1, 1)
(1, 1, 2)
(1, 2, 1)
(1, 3)
(2, 1, 1)
(2, 2)
(3, 1)

Note that different sequences are counted as different combinations.

Therefore the output is 7.
```

Follow up:

What if negative numbers are allowed in the given array?

How does it change the problem?

What limitation we need to add to the question to allow negative numbers?

Companies

Google   Facebook   Snapchat

Topics

Dynamic Programming

39. Combination Sum (Medium)

Given a set of candidate numbers (C) (without duplicates) and a target number (T), find all unique combinations in C where the candidate numbers sums to T.

The same repeated number may be chosen from C unlimited number of times.

Note:

- All numbers (including target) will be positive integers.
- The solution set must not contain duplicate combinations.

For example, given candidate set `[2, 3, 6, 7]` and target `7`,
A solution set is:

```
[
  [7],
  [2, 2, 3]
]
```

Companies

Uber Snapchat

Topics

Array Backtracking

77. Combinations (Medium)

Given two integers n and k, return all possible combinations of k numbers out of 1 ... n.

For example,
If n = 4 and k = 2, a solution is:

```
[
  [2,4],
  [3,4],
  [2,3],
  [1,2],
  [1,3],
  [1,4],
]
```

Companies

Topics

Backtracking

165. Compare Version Numbers (Medium)

Compare two version numbers version1 and version2.
If version1 > version2 return 1, if version1 < version2 return -1, otherwise return 0.

You may assume that the version strings are non-empty and contain only digits and the
`.` character.
The `.` character does not represent a decimal point and is used to separate number
sequences.
For instance, `2.5` is not "two and a half" or "half way to version three", it is the fifth
second-level revision of the second first-level revision.

Here is an example of version numbers ordering:

```
0.1 < 1.1 < 1.2 < 13.37
```

Companies

Microsoft   Apple

Topics

String

106. Construct Binary Tree from Inorder and Postorder Traversal (Medium)

Given inorder and postorder traversal of a tree, construct the binary tree.

Note:

You may assume that duplicates do not exist in the tree.

Companies

Microsoft

Topics

Array   Tree   Depth-first Search

105. Construct Binary Tree from Preorder and Inorder Traversal (Medium)

Given preorder and inorder traversal of a tree, construct the binary tree.

Note:

You may assume that duplicates do not exist in the tree.

Companies

(Bloomberg)

Topics

(Array) (Tree) (Depth-first Search)

11. Container With Most Water (Medium)

Given n non-negative integers a_1, a_2, ..., a_n, where each represents a point at coordinate (i, a_i). n vertical lines are drawn such that the two endpoints of line i is at (i, a_i) and $(i, 0)$. Find two lines, which together with x-axis forms a container, such that the container contains the most water.

Note: You may not slant the container and n is at least 2.

Companies

Bloomberg

Topics

Array   Two Pointers

219. Contains Duplicate II (Easy)

Given an array of integers and an integer k, find out whether there are two distinct indices i and j in the array such that nums[i] = nums[j]   and the absolute difference between i and j is at most k.

Companies

Airbnb   Palantir

Topics

Array   Hash Table

220. Contains Duplicate III (Medium)

Given an array of integers, find out whether there are two distinct indices i and j in the array such that the absolute difference between nums[i] and nums[j] is at most t and the absolute difference between i and j is at most k.

Companies

Airbnb Palantir

Topics

Binary Search Tree

217. Contains Duplicate (Easy)

Given an array of integers, find if the array contains any duplicates. Your function should return true if any value appears at least twice in the array, and it should return false if every element is distinct.

Companies

Airbnb Yahoo Palantir

Topics

Array Hash Table

405. Convert a Number to Hexadecimal (Easy)

Given an integer, write an algorithm to convert it to hexadecimal. For negative integer, two's complement (https://en.wikipedia.org/wiki/Two%27s_complement) method is used.

Note:

1. All letters in hexadecimal (a-f) must be in lowercase.
2. The hexadecimal string must not contain extra leading 0s. If the number is zero, it is represented by a single zero character '0'; otherwise, the first character in the hexadecimal string will not be the zero character.
3. The given number is guaranteed to fit within the range of a 32-bit signed integer.
4. You must not use   any method provided by the library   which converts/formats the number to hex directly.

Example 1:

```
Input:
26

Output:
"1a"
```

Example 2:

```
Input:
-1

Output:
"ffffffff"
```

Companies

Topics

Bit Manipulation

108. Convert Sorted Array to Binary Search Tree (Easy)

Given an array where elements are sorted in ascending order, convert it to a height balanced BST.

Companies

Airbnb

Topics

Tree   Depth-first Search

109. Convert Sorted List to Binary Search Tree (Medium)

Given a singly linked list where elements are sorted in ascending order, convert it to a height balanced BST.

Companies

Zenefits

Topics

Linked List   Depth-first Search

138. Copy List with Random Pointer (Medium)

A linked list is given such that each node contains an additional random pointer which could point to any node in the list or null.

Return a deep copy of the list.

Companies

Microsoft   Amazon   Bloomberg   Uber

Topics

Hash Table   Linked List

38. Count and Say (Easy)

The count-and-say sequence is the sequence of integers with the first five terms as following:

```
1.      1
2.      11
3.      21
4.      1211
5.      111221
```

`1` is read off as `"one 1"` or `11`.

`11` is read off as `"two 1s"` or `21`.

`21` is read off as `"one 2, then one 1"` or `1211`.

Given an integer n, generate the nth term of the count-and-say sequence.

Note: Each term of the sequence of integers will be represented as a string.

Example 1:

```
Input: 1
Output: "1"
```

Example 2:

```
Input: 4
Output: "1211"
```

Companies

Facebook

Topics

String

222. Count Complete Tree Nodes (Medium)

Given a complete binary tree, count the number of nodes.

Definition of a complete binary tree from   Wikipedia
(https://en.wikipedia.org/wiki/Binary_tree#T   ypes_of_binary_trees) :
In a complete binary tree every level, except possibly the last, is completely filled, and
all nodes in the last level are as far left as possible. It can have between 1 and 2^h
nodes inclusive at the last level h.

Companies

Topics

Binary Search   Tree

357. Count Numbers with Unique Digits (Medium)

Given a non-negative integer n, count all numbers with unique digits, x, where $0 \le x < 10^n$.

Example:

Given n = 2, return 91. (The answer should be the total numbers in the range of $0 \le x <$ 100, excluding `[11,22,33,44,55,66,77,88,99]`)

Companies

Google

Topics

Math   Dynamic Programming   Backtracking

327. Count of Range Sum (Hard)

Given an integer array `nums`, return the number of range sums that lie in `[lower, upper]` inclusive.

Range sum `S(i, j)` is defined as the sum of the elements in `nums` between indices `i` and `j` (`i ≤ j`), inclusive.

Note:

A naive algorithm of $O(n^2)$ is trivial. You MUST do better than that.

Example:

Given nums = `[-2, 5, -1]`, lower = `-2`, upper = `2`,

Return `3`.

The three ranges are : `[0, 0]`, `[2, 2]`, `[0, 2]` and their respective sums are: `-2`, `-1`, `2`.

Companies

Google

Topics

Divide and Conquer   Binary Search Tree

315. Count of Smaller Numbers After Self (Hard)

You are given an integer array nums and you have to return a new counts array. The counts array has the property where `counts[i]` is the number of smaller elements to the right of `nums[i]`.

Example:

```
Given nums = [5, 2, 6, 1]

To the right of 5 there are 2 smaller elements (2 and 1).
To the right of 2 there is only 1 smaller element (1).
To the right of 6 there is 1 smaller element (1).
To the right of 1 there is 0 smaller element.
```

Return the array `[2, 1, 1, 0]`.

Companies

Google

Topics

Divide and Conquer   Binary Indexed Tree   Segment Tree   Binary Search Tree

204. Count Primes (Easy)

Description:

Count the number of prime numbers less than a non-negative number, n.

Companies

(Microsoft) (Amazon)
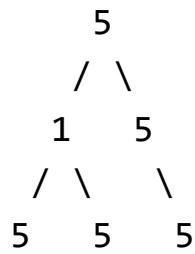
Topics

(Hash Table) (Math)

250. Count Univalue Subtrees (Medium)

Given a binary tree, count the number of uni-value subtrees.

A Uni-value subtree means all nodes of the subtree have the same value.

For example:
Given binary tree,

```
        5
       / \
      1   5
     / \   \
    5   5   5
```

return 4.

Companies

Topics

Tree

338. Counting Bits (Medium)

Given a non negative integer number `num`. For every numbers i in the range 0 ≤ i ≤ `num` calculate the number of 1's in their binary representation and return them as an array.

Example:
For `num = 5` you should return `[0,1,1,2,1,2]`.

Follow up:

- It is very easy to come up with a solution with run time O(n*sizeof(integer)) . But can you do it in linear time O(n) /possibly in a single pass?
- Space complexity should be O(n).
- Can you do it like a boss? Do it without using any builtin function like `__builtin_popcount` in c++ or in any other language.

Companies

Topics

Dynamic Programming   Bit Manipulation

210. Course Schedule II (Medium)

There are a total of n courses you have to take, labeled from `0` to `n - 1`.

Some courses may have prerequisites, for example to take course 0 you have to first take course 1, which is expressed as a pair: `[0,1]`

Given the total number of courses and a list of prerequisite pairs , return the ordering of courses you should take to finish all courses.

There may be multiple correct orders, you just need to return one of them. If it is impossible to finish all courses, return an empty array.

For example:

```
2, [[1,0]]
```

There are a total of 2 courses to take. To take course 1 you should have finished course 0. So the correct course order is `[0,1]`

```
4, [[1,0],[2,0],[3,1],[3,2]]
```

There are a total of 4 courses to take. To take course 3 you should have finished both courses 1 and 2. Both courses 1 and 2 should be taken after you finished course 0. So one correct course order is `[0,1,2,3]`. Another correct ordering is `[0,2,1,3]`.

Note:

1. The input prerequisites is a graph represented by a list of edges , not adjacency matrices. Read more about how a graph is represented (https://www.khanacademy.org/computing/computer-science/algorithms/graph-representation/a/representing-graphs).
2. You may assume that there are no duplicate edges in the input prerequisites.

Hints:

1. This problem is equivalent to finding the topological order in a directed graph. If a cycle exists, no topological ordering exists and therefore it will be impossible to take all courses.
2. Topological Sort via DFS (https://class.coursera.org/algo-003/lecture/52) - A great video tutorial (21 minutes) on Coursera explaining the basic concepts of Topological Sort.
3. Topological sort could also be done via BFS (https://en.wikipedia.org/wiki/Topological_sorting#Algorithms).

Companies

Facebook   Zenefits

Topics

Depth-first Search   Breadth-first Search   Graph   Topological Sort

207. Course Schedule (Medium)

There are a total of n courses you have to take, labeled from `0` to `n - 1`.

Some courses may have prerequisites, for example to take course 0 you have to first take course 1, which is expressed as a pair: `[0,1]`

Given the total number of courses and a list of prerequisite pairs , is it possible for you to finish all courses?

For example:

```
2, [[1,0]]
```

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

```
2, [[1,0],[0,1]]
```

There are a total of 2 courses to take. To take course 1 you should have finished course 0, and to take course 0 you should also have finished course 1. So it is impossible.

Note:

1. The input prerequisites is a graph represented by a list of edges , not adjacency matrices. Read more about how a graph is represented (https://www.khanacademy.org/computing/computer-science/algorithms/graph-representation/a/representing-graphs).
2. You may assume that there are no duplicate edges in the input prerequisites.

Hints:

1. This problem is equivalent to finding if a cycle exists in a directed graph. If a cycle exists, no topological ordering exists and therefore it will be impossible to take all courses.

2. Topological Sort via DFS (https://class.coursera.org/algo-003/lecture/52) - A great video tutorial (21 minutes) on Coursera explaining the basic concepts of Topological Sort.

3. Topological sort could also be done via BFS (https://en.wikipedia.org/wiki/Topological_sorting#Algorithms).

Companies

Uber   Apple   Zenefits   Yelp

Topics

Depth-first Search   Breadth-first Search   Graph   Topological Sort

321. Create Maximum Number (Hard)

Given two arrays of length `m` and `n` with digits `0-9` representing two numbers. Create the maximum number of length `k <= m + n` from digits of the two. The relative order of the digits from the same array must be preserved. Return an array of the `k` digits. You should try to optimize your time and space complexity.

Example 1:

nums1 = `[3, 4, 6, 5]`
nums2 = `[9, 1, 2, 5, 8, 3]`
k = `5`
return `[9, 8, 6, 5, 3]`

Example 2:

nums1 = `[6, 7]`
nums2 = `[6, 0, 4]`
k = `5`
return `[6, 7, 6, 0, 4]`

Example 3:

nums1 = `[3, 9]`
nums2 = `[8, 9]`
k = `3`
return `[9, 8, 9]`

Companies

Google

Topics

Dynamic Programming   Greedy

352. Data Stream as Disjoint Intervals (Hard)

Given a data stream input of non-negative integers a_1, a_2, ..., a_n, ..., summarize the numbers seen so far as a list of disjoint intervals.

For example, suppose the integers from the data stream are 1, 3, 7, 2, 6, ..., then the summary will be:

```
[1, 1]
[1, 1], [3, 3]
[1, 1], [3, 3], [7, 7]
[1, 3], [7, 7]
[1, 3], [6, 7]
```

Follow up:

What if there are lots of merges and the number of disjoint intervals are small compared to the data stream's size?

Companies

Topics

Binary Search Tree

394. Decode String (Medium)

Given an encoded string, return it's decoded string.

The encoding rule is: `k[encoded_string]`, where the encoded_string inside the square brackets is being repeated exactly k times. Note that k is guaranteed to be a positive integer.

You may assume that the input string is always valid; No extra white spaces, square brackets are well-formed, etc.

Furthermore, you may assume that the original data does not contain any digits and that digits are only for those repeat numbers, k. For example, there won't be input like `3a` or `2[4]`.

Examples:

```
s = "3[a]2[bc]", return "aaabcbc".
s = "3[a2[c]]", return "accaccacc".
s = "2[abc]3[cd]ef", return "abcabccdcdcdef".
```

Companies

Google Yelp Coupang

Topics

Stack Depth-first Search

91. Decode Ways (Medium)

A message containing letters from `A-Z` is being encoded to numbers using the following mapping:

```
'A' -> 1
'B' -> 2
...
'Z' -> 26
```

Given an encoded message containing digits, determine the total number of ways to decode it.

For example,
Given encoded message `"12"`, it could be decoded as `"AB"` (1 2) or `"L"` (12).

The number of ways decoding `"12"` is 2.

Companies

Facebook   Microsoft   Uber

Topics

String   Dynamic Programming

237. Delete Node in a Linked List (Easy)

Write a function to delete a node (except the tail) in a singly linked list, given only access to that node.

Supposed the linked list is `1 -> 2 -> 3 -> 4` and you are given the third node with value `3`, the linked list should become `1 -> 2 -> 4` after calling your function.

Companies

Microsoft   Apple   Adobe

Topics

Linked List

362. Design Hit Counter (Medium)

Design a hit counter which counts the number of hits received in the past 5 minutes.

Each function accepts a timestamp parameter (in seconds granularity) and you may assume that calls are being made to the system in chronological order (ie, the timestamp is monotonically increasing). You may assume that the earliest timestamp starts at 1.

It is possible that several hits arrive roughly at the same time.

Example:

```
HitCounter counter = new HitCounter();

// hit at timestamp 1.
counter.hit(1);

// hit at timestamp 2.
counter.hit(2);

// hit at timestamp 3.
counter.hit(3);

// get hits at timestamp 4, should return 3.
counter.getHits(4);

// hit at timestamp 300.
counter.hit(300);

// get hits at timestamp 300, should return 4.
counter.getHits(300);

// get hits at timestamp 301, should return 3.
counter.getHits(301);
```

Follow up:

What if the number of hits per second could be very large? Does your design scale?

Companies

Google   Dropbox

Topics

Design

379. Design Phone Directory (Medium)

Design a Phone Directory which supports the following operations:

1. `get` : Provide a number which is not assigned to anyone.
2. `check` : Check if a number is available or not.
3. `release` : Recycle or release a number.

Example:

```
// Init a phone directory containing a total of 3 numbers: 0, 1, and
2.
PhoneDirectory directory = new PhoneDirectory(3);

// It can return any available phone number. Here we assume it retur
ns 0.
directory.get();

// Assume it returns 1.
directory.get();

// The number 2 is available, so return true.
directory.check(2);

// It returns 2, the only number that is left.
directory.get();

// The number 2 is no longer available, so return false.
directory.check(2);

// Release number 2 back to the pool.
directory.release(2);

// Number 2 is available again, return true.
directory.check(2);
```

Companies

Google

Topics

Linked List   Design

353. Design Snake Game (Medium)

Design a Snake game (https://en.wikipedia.org/wiki/Snake_(video_game)) that is played on a device with screen size = width x height. Play the game online (http://patorjk.com/games/snake/) if you are not familiar with the game.

The snake is initially positioned at the top left corner (0,0) with length = 1 unit.

You are given a list of food's positions in row-column order. When a snake eats the food, its length and the game's score both increase by 1.

Each food appears one by one on the screen. For example, the second food will not appear until the first food was eaten by the snake.

When a food does appear on the screen, it is guaranteed that it will not appear on a block occupied by the snake.

Example:

Given width = 3, height = 2, and food = [[1,2],[0,1]].

Snake snake = new Snake(width, height, food);

Initially the snake appears at position (0,0) and the food at (1,2).

```
|S| | |
| | |F|
```

snake.move("R"); -> Returns 0

```
| |S| |
| | |F|
```

snake.move("D"); -> Returns 0

```
| | | |
| |S|F|
```

snake.move("R"); -> Returns 1 (Snake eats the first food and right after that, the second food appears at (0,1))

```
| |F| |
| |S|S|
```

snake.move("U"); -> Returns 1

```
| |F|S|
| | |S|
```

snake.move("L"); -> Returns 2 (Snake eats the second food)

```
| |S|S|
| | |S|
```

snake.move("U"); -> Returns -1 (Game over because snake collides wit
```

```
h border)
```

Companies

Google

Topics

Design   Queue

348. Design Tic-Tac-Toe (Medium)

Design a Tic-tac-toe game that is played between two players on a n x n grid.

You may assume the following rules:

1. A move is guaranteed to be valid and is placed on an empty block.
2. Once a winning condition is reached, no more moves is allowed.
3. A player who succeeds in placing n of their marks in a horizontal, vertical, or diagonal row wins the game.

Example:

```
Given n = 3, assume that player 1 is "X" and player 2 is "O" in the
 board.

TicTacToe toe = new TicTacToe(3);

toe.move(0, 0, 1); -> Returns 0 (no one wins)
|X| | |
| | | |    // Player 1 makes a move at (0, 0).
| | | |

toe.move(0, 2, 2); -> Returns 0 (no one wins)
|X| |O|
| | | |    // Player 2 makes a move at (0, 2).
| | | |

toe.move(2, 2, 1); -> Returns 0 (no one wins)
|X| |O|
| | | |    // Player 1 makes a move at (2, 2).
| | |X|

toe.move(1, 1, 2); -> Returns 0 (no one wins)
|X| |O|
| |O| |    // Player 2 makes a move at (1, 1).
| | |X|

toe.move(2, 0, 1); -> Returns 0 (no one wins)
|X| |O|
| |O| |    // Player 1 makes a move at (2, 0).
|X| |X|

toe.move(1, 0, 2); -> Returns 0 (no one wins)
|X| |O|
|O|O| |    // Player 2 makes a move at (1, 0).
|X| |X|

toe.move(2, 1, 1); -> Returns 1 (player 1 wins)
```

```
|X| |O|
|O|O| |      // Player 1 makes a move at (2, 1).
|X|X|X|
```

Follow up:

Could you do better than O(n^2) per `move()` operation?

Companies

Google   Microsoft

Topics

Design

355. Design Twitter (Medium)

Design a simplified version of Twitter where users can post tweets, follow/unfollow another user and is able to see the 10 most recent tweets in the user's news feed. Your design should support the following methods:

1. postTweet(userId, tweetId) : Compose a new tweet.
2. getNewsFeed(userId) : Retrieve the 10 most recent tweet ids in the user's news feed. Each item in the news feed must be posted by users who the user followed or by the user herself. Tweets must be ordered from most recent to least recent.
3. follow(followerId, followeeId)   : Follower follows a followee.
4. unfollow(followerId, followeeId)   : Follower unfollows a followee.

Example:

```
Twitter twitter = new Twitter();

// User 1 posts a new tweet (id = 5).
twitter.postTweet(1, 5);

// User 1's news feed should return a list with 1 tweet id -> [5].
twitter.getNewsFeed(1);

// User 1 follows user 2.
twitter.follow(1, 2);

// User 2 posts a new tweet (id = 6).
twitter.postTweet(2, 6);

// User 1's news feed should return a list with 2 tweet ids -> [6,
 5].
// Tweet id 6 should precede tweet id 5 because it is posted after t
weet id 5.
twitter.getNewsFeed(1);

// User 1 unfollows user 2.
twitter.unfollow(1, 2);

// User 1's news feed should return a list with 1 tweet id -> [5],
// since user 1 is no longer following user 2.
twitter.getNewsFeed(1);
```

Companies

Amazon   Twitter

Topics

Hash Table   Heap   Design

241. Different Ways to Add Parentheses (Medium)

Given a string of numbers and operators, return all possible results from computing all the different possible ways to group numbers and operators. The valid operators are `+`, `-` and `*`.

Example 1

Input: `"2-1-1"`.

```
((2-1)-1) = 0
(2-(1-1)) = 2
```

Output: `[0, 2]`

Example 2

Input: `"2*3-4*5"`

```
(2*(3-(4*5))) = -34
((2*3)-(4*5)) = -14
((2*(3-4))*5) = -10
(2*((3-4)*5)) = -10
(((2*3)-4)*5) = 10
```

Output: `[-34, -14, -10, -10, 10]`

Companies

Topics

Divide and Conquer

115. Distinct Subsequences (Hard)

Given a string S and a string T, count the number of distinct subsequences of S which equals T.

A subsequence of a string is a new string which is formed from the original string by deleting some (can be none) of the characters without disturbing the relative positions of the remaining characters. (ie, `"ACE"` is a subsequence of `"ABCDE"` while `"AEC"` is not).

Here is an example:
S = `"rabbbit"`, T = `"rabbit"`

Return `3`.

Companies

Topics

String   Dynamic Programming

29. Divide Two Integers (Medium)

Divide two integers without using multiplication, division and mod operator.

If it is overflow, return MAX_INT.

Companies

Topics

Math   Binary Search

174. Dungeon Game (Hard)

The demons had captured the princess (P) and imprisoned her in the bottom-right corner of a dungeon. The dungeon consists of M x N rooms laid out in a 2D grid. Our valiant knight (K) was initially positioned in the top-left room and must fight his way through the dungeon to rescue the princess.

The knight has an initial health point represented by a positive integer. If at any point his health point drops to 0 or below, he dies immediately.

Some of the rooms are guarded by demons, so the knight loses health (negative integers) upon entering these rooms; other rooms are either empty (0's) or contain magic orbs that increase the knight's health (positive integers).

In order to reach the princess as quickly as possible, the knight decides to move only rightward or downward in each step.

Write a function to determine the knight's minimum initial health so that he is able to rescue the princess.

For example, given the dungeon below, the initial health of the knight must be at least 7 if he follows the optimal path `RIGHT-> RIGHT -> DOWN -> DOWN`.

-2 (K)	-3	3
-5	-10	1
10	30	-5 (P)

Notes:

- The knight's health has no upper bound.

- Any room can contain threats or power-ups, even the first room the knight enters and the bottom-right room where the princess is imprisoned.

Companies

Microsoft

Topics

Binary Search   Dynamic Programming

72. Edit Distance (Hard)

Given two words word1 and word2, find the minimum number of steps required to convert word1 to word2. (each operation is counted as 1 step.)

You have the following 3 operations permitted on a word:

a) Insert a character
b) Delete a character
c) Replace a character

Companies

Topics

String   Dynamic Programming

390. Elimination Game (Medium)

There is a list of sorted integers from 1 to n. Starting from left to right, remove the first number and every other number afterward until you reach the end of the list.

Repeat the previous step again, but this time from right to left, remove the right most number and every other number from the remaining numbers.

We keep repeating the steps again, alternating left to right and right to left, until a single number remains.

Find the last number that remains starting with a list of length n.

Example:

```
Input:
n = 9,
1 2 3 4 5 6 7 8 9
2 4 6 8
2 6
6

Output:
6
```

Companies

Topics

271. Encode and Decode Strings (Medium)

Design an algorithm to encode a list of strings to a string . The encoded string is then sent over the network and is decoded back to the original list of strings.

Machine 1 (sender) has the function:

```
string encode(vector<string> strs) {
  // ... your code
  return encoded_string;
}
```

Machine 2 (receiver) has the function:

```
vector<string> decode(string s) {
  //... your code
  return strs;
}
```

So Machine 1 does:

```
string encoded_string = encode(strs);
```

and Machine 2 does:

```
vector<string> strs2 = decode(encoded_string);
```

`strs2` in Machine 2 should be the same as `strs` in Machine 1.

Implement the `encode` and `decode` methods.

Note:

- The string may contain any possible characters out of 256 valid ascii characters. Your algorithm should be generalized enough to work on any possible characters.
- Do not use class member/global/static variables to store states. Your encode and decode algorithms should be stateless.

- Do not rely on any library method such as `eval` or serialize methods. You should implement your own encode/decode algorithm.

Companies

Google

Topics

String

399. Evaluate Division (Medium)

Equations are given in the format `A / B = k`, where `A` and `B` are variables represented as strings, and `k` is a real number (floating point number). Given some queries, return the answers. If the answer does not exist, return `-1.0`.

Example:
Given `a / b = 2.0, b / c = 3.0`.
queries are: `a / c = ?, b / a = ?, a / e = ?, a / a = ?, x / x = ?` .
return `[6.0, 0.5, -1.0, 1.0, -1.0 ]`.

The input is: `vector<pair<string, string>> equations, vector<double>& values, vector<pair<string, string>> queries` , where `equations.size() == values.size()`, and the values are positive. This represents the equations. Return `vector<double>`.

According to the example above:

```
equations = [ ["a", "b"], ["b", "c"] ],
values = [2.0, 3.0],
queries = [ ["a", "c"], ["b", "a"], ["a", "e"], ["a", "a"], ["x",
 "x"] ].
```

The input is always valid. You may assume that evaluating the queries will result in no division by zero and there is no contradiction.

Companies

Google

Topics

Graph

150. Evaluate Reverse Polish Notation (Medium)

Evaluate the value of an arithmetic expression in Reverse Polish Notation (https://en.wikipedia.org/wiki/Reverse_Polish_notation).

Valid operators are `+`, `-`, `*`, `/`. Each operand may be an integer or another expression.

Some examples:

```
["2", "1", "+", "3", "*"] -> ((2 + 1) * 3) -> 9
["4", "13", "5", "/", "+"] -> (4 + (13 / 5)) -> 6
```

Companies

LinkedIn

Topics

Stack

171. Excel Sheet Column Number (Easy)

Related to question Excel Sheet Column Title (https://leetcode.com/problems/excel-sheet-column-title/)

Given a column title as appear in an Excel sheet, return its corresponding column number.

For example:

```
A -> 1
B -> 2
C -> 3
...
Z -> 26
AA -> 27
AB -> 28
```

Companies

Microsoft Uber

Topics

Math

168. Excel Sheet Column Title (Easy)

Given a positive integer, return its corresponding column title as appear in an Excel sheet.

For example:

```
1 -> A
2 -> B
3 -> C
...
26 -> Z
27 -> AA
28 -> AB
```

Companies

Facebook   Microsoft   Zenefits

Topics

Math

282. Expression Add Operators (Hard)

Given a string that contains only digits `0-9` and a target value, return all possibilities to add binary operators (not unary) `+`, `-`, or `*` between the digits so they evaluate to the target value.

Examples:

```
"123", 6 -> ["1+2+3", "1*2*3"]
"232", 8 -> ["2*3+2", "2+3*2"]
"105", 5 -> ["1*0+5","10-5"]
"00", 0 -> ["0+0", "0-0", "0*0"]
"3456237490", 9191 -> []
```

Companies

Google   Facebook

Topics

Divide and Conquer

254. Factor Combinations (Medium)

Numbers can be regarded as product of its factors. For example,

```
8 = 2 x 2 x 2;
  = 2 x 4.
```

Write a function that takes an integer n and return all possible combinations of its factors.

Note:

1. You may assume that n is always positive.
2. Factors should be greater than 1 and less than n.

Examples:
input: 1
output:

```
[]
```

input: 37
output:
```
[]
```

input: 12
output:

```
[
  [2, 6],
  [2, 2, 3],
  [3, 4]
]
```

input: 32
output:

```
[
  [2, 16],
  [2, 2, 8],
  [2, 2, 2, 4],
  [2, 2, 2, 2, 2],
  [2, 4, 4],
  [4, 8]
]
```

Companies

Uber   LinkedIn

Topics

Backtracking

172. Factorial Trailing Zeroes (Easy)

Given an integer n, return the number of trailing zeroes in n!.

Note: Your solution should be in logarithmic time complexity.

Companies

(Bloomberg)

Topics

(Math)

373. Find K Pairs with Smallest Sums (Medium)

You are given two integer arrays nums1 and nums2 sorted in ascending order and an integer k.

Define a pair (u,v) which consists of one element from the first array and one element from the second array.

Find the k pairs $(u_1,v_1),(u_2,v_2) ...(u_k,v_k)$ with the smallest sums.

Example 1:

```
Given nums1 = [1,7,11], nums2 = [2,4,6],  k = 3

Return: [1,2],[1,4],[1,6]

The first 3 pairs are returned from the sequence:
[1,2],[1,4],[1,6],[7,2],[7,4],[11,2],[7,6],[11,4],[11,6]
```

Example 2:

```
Given nums1 = [1,1,2], nums2 = [1,2,3],  k = 2

Return: [1,1],[1,1]

The first 2 pairs are returned from the sequence:
[1,1],[1,1],[1,2],[2,1],[1,2],[2,2],[1,3],[1,3],[2,3]
```

Example 3:

```
Given nums1 = [1,2], nums2 = [3],  k = 3

Return: [1,3],[2,3]

All possible pairs are returned from the sequence:
[1,3],[2,3]
```

Companies

Google Uber

Topics

Heap

366. Find Leaves of Binary Tree (Medium)

Given a binary tree, collect a tree's nodes as if you were doing this: Collect and remove all leaves, repeat until the tree is empty.

Example:
Given binary tree

```
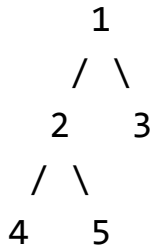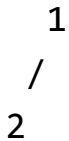        1
       / \
      2   3
     / \
    4   5
```

Returns `[4, 5, 3]`, `[2]`, `[1]`.

Explanation:

1. Removing the leaves `[4, 5, 3]` would result in this tree:

```
        1
       /
      2
```

2. Now removing the leaf `[2]` would result in this tree:

```
        1
```

3. Now removing the leaf `[1]` would result in the empty tree:

```
        []
```

Returns `[4, 5, 3]`, `[2]`, `[1]`.

Companies

(LinkedIn)

Topics

Tree   Depth-first Search

295. Find Median from Data Stream (Hard)

Median is the middle value in an ordered integer list. If the size of the list is even, there is no middle value. So the median is the mean of the two middle value.

Examples:

[2,3,4] , the median is 3

[2,3], the median is (2 + 3) / 2 = 2.5

Design a data structure that supports the following two operations:

- void addNum(int num) - Add a integer number from the data stream to the data structure.
- double findMedian() - Return the median of all elements so far.

For example:

```
addNum(1)
addNum(2)
findMedian() -> 1.5
addNum(3)
findMedian() -> 2
```

Companies

Google

Topics

Heap   Design

154. Find Minimum in Rotated Sorted Array II (Hard)

Follow up for "Find Minimum in Rotated Sorted Array":

What if duplicates are allowed?

Would this affect the run-time complexity? How and why?

Suppose an array sorted in ascending order is rotated at some pivot unknown to you beforehand.

(i.e., `0 1 2 4 5 6 7` might become `4 5 6 7 0 1 2`).

Find the minimum element.

The array may contain duplicates.

Companies

Topics

Array   Binary Search

153. Find Minimum in Rotated Sorted Array (Medium)

Suppose an array sorted in ascending order is rotated at some pivot unknown to you beforehand.

(i.e., `0 1 2 4 5 6 7` might become `4 5 6 7 0 1 2`).

Find the minimum element.

You may assume no duplicate exists in the array.

Companies

Microsoft

Topics

Array   Binary Search

162. Find Peak Element (Medium)

A peak element is an element that is greater than its neighbors.

Given an input array where `num[i] ≠ num[i+1]`, find a peak element and return its index.

The array may contain multiple peaks, in that case return the index to any one of the peaks is fine.

You may imagine that `num[-1] = num[n] = -∞`.

For example, in array `[1, 2, 3, 1]`, 3 is a peak element and your function should return the index number 2.

Note:

Your solution should be in logarithmic complexity.

Companies

Google Microsoft

Topics

Array Binary Search

277. Find the Celebrity (Medium)

Suppose you are at a party with `n` people (labeled from `0` to `n - 1`) and among them, there may exist one celebrity. The definition of a celebrity is that all the other `n - 1` people know him/her but he/she does not know any of them.

Now you want to find out who the celebrity is or verify that there is not one. The only thing you are allowed to do is to ask questions like: "Hi, A. Do you know B?" to get information of whether A knows B. You need to find out the celebrity (or verify there is not one) by asking as few questions as possible (in the asymptotic sense).

You are given a helper function `bool knows(a, b)` which tells you whether A knows B. Implement a function `int findCelebrity(n)`, your function should minimize the number of calls to `knows`.

Note: There will be exactly one celebrity if he/she is in the party. Return the celebrity's label if there is a celebrity in the party. If there is no celebrity, return `-1`.

Companies

Topics

389. Find the Difference (Easy)

Given two strings s and t which consist of only lowercase letters.

String t is generated by random shuffling string s and then add one more letter at a random position.

Find the letter that was added in t.

Example:

```
Input:
s = "abcd"
t = "abcde"

Output:
e

Explanation:
'e' is the letter that was added.
```

Companies

Google

Topics

Hash Table   Bit Manipulation

287. Find the Duplicate Number (Medium)

Given an array nums containing n + 1 integers where each integer is between 1 and n (inclusive), prove that at least one duplicate number must exist. Assume that there is only one duplicate number, find the duplicate one.

Note:

1. You must not modify the array (assume the array is read only).
2. You must use only constant, O(1) extra space.
3. Your runtime complexity should be less than `O(n²)` .
4. There is only one duplicate number in the array, but it could be repeated more than once.

Companies

Bloomberg

Topics

Array   Two Pointers   Binary Search

278. First Bad Version (Easy)

You are a product manager and currently leading a team to develop a new product. Unfortunately, the latest version of your product fails the quality check. Since each version is developed based on the previous version, all the versions after a bad version are also bad.

Suppose you have `n` versions `[1, 2, ..., n]` and you want to find out the first bad one, which causes all the following ones to be bad.

You are given an API `bool isBadVersion(version)` which will return whether `version` is bad. Implement a function to find the first bad version. You should minimize the number of calls to the API.

Companies

Facebook

Topics

Binary Search

41. First Missing Positive (Hard)

Given an unsorted integer array, find the first missing positive integer.

For example,
Given `[1,2,0]` return `3`,
and `[3,4,-1,1]` return `2`.

Your algorithm should run in O(n) time and uses constant space.

Companies

Topics

Array

387. First Unique Character in a String (Easy)

Given a string, find the first non-repeating character in it and return it's index. If it doesn't exist, return -1.

Examples:

```
s = "leetcode"
return 0.

s = "loveleetcode",
return 2.
```

Note: You may assume the string contain only lowercase letters.

Companies

Microsoft   Amazon   Bloomberg

Topics

412. Fizz Buzz (Easy)

Write a program that outputs the string representation of numbers from 1 to n.

But for multiples of three it should output "Fizz" instead of the number and for the multiples of five output "Buzz". For numbers which are multiples of both three and five output "FizzBuzz".

Example:

```
n = 15,

Return:
[
    "1",
    "2",
    "Fizz",
    "4",
    "Buzz",
    "Fizz",
    "7",
    "8",
    "Fizz",
    "Buzz",
    "11",
    "Fizz",
    "13",
    "14",
    "FizzBuzz"
]
```

Companies

Topics

251. Flatten 2D Vector (Medium)

Implement an iterator to flatten a 2d vector.

For example,
Given 2d vector =

```
[
  [1,2],
  [3],
  [4,5,6]
]
```

By calling next repeatedly until hasNext returns false, the order of elements returned by next should be: `[1,2,3,4,5,6]`.

Follow up:
As an added challenge, try to code it using only iterators in C++ (http://www.cplusplus.com/reference/iterator/iterator/) or iterators in Java (http://docs.oracle.com/javase/7/docs/api/java/util/Iterator.html).

Companies

Google   Twitter   Airbnb   Zenefits

Topics

Design

114. Flatten Binary Tree to Linked List (Medium)

Given a binary tree, flatten it to a linked list in-place.

For example,
Given

```
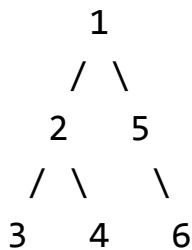      1
     / \
    2   5
   / \   \
  3   4   6
```

The flattened tree should look like:

```
  1
   \
    2
     \
      3
       \
        4
         \
          5
           \
            6
```

Hints:

If you notice carefully in the flattened tree, each node's right child points to the next node of a pre-order traversal.

Companies

Microsoft

Topics

Tree Depth-first Search

341. Flatten Nested List Iterator (Medium)

Given a nested list of integers, implement an iterator to flatten it.

Each element is either an integer, or a list -- whose elements may also be integers or other lists.

Example 1:
Given the list `[[1,1],2,[1,1]]`,

By calling next repeatedly until hasNext returns false, the order of elements returned by next should be: `[1,1,2,1,1]`.

Example 2:
Given the list `[1,[4,[6]]]`,

By calling next repeatedly until hasNext returns false, the order of elements returned by next should be: `[1,4,6]`.

Companies

(Google) (Facebook) (Twitter)

Topics

(Stack) (Design)

294. Flip Game II (Medium)

You are playing the following Flip Game with your friend: Given a string that contains only these two characters: `+` and `-`, you and your friend take turns to flip two consecutive `"++"` into `"--"`. The game ends when a person can no longer make a move and therefore the other person will be the winner.

Write a function to determine if the starting player can guarantee a win.

For example, given `s = "++++"`, return true. The starting player can guarantee a win by flipping the middle `"++"` to become `"+--+"`.

Follow up:
Derive your algorithm's runtime complexity.

Companies

(Google)

Topics

(Backtracking)

293. Flip Game (Easy)

You are playing the following Flip Game with your friend: Given a string that contains only these two characters: `+` and `-`, you and your friend take turns to flip two consecutive `"++"` into `"--"`. The game ends when a person can no longer make a move and therefore the other person will be the winner.

Write a function to compute all possible states of the string after one valid move.

For example, given `s = "++++"`, after one move, it may become one of the following states:

```
[
    "--++",
    "+--+",
    "++--"
]
```

If there is no valid move, return an empty list `[]`.

Companies

(Google)

Topics

(String)

166. Fraction to Recurring Decimal (Medium)

Given two integers representing the numerator and denominator of a fraction, return the fraction in string format.

If the fractional part is repeating, enclose the repeating part in parentheses.

For example,

- Given numerator = 1, denominator = 2, return "0.5".
- Given numerator = 2, denominator = 1, return "2".
- Given numerator = 2, denominator = 3, return "0.(6)".

Companies

Google   IXL

Topics

Hash Table   Math

403. Frog Jump (Hard)

A frog is crossing a river. The river is divided into x units and at each unit there may or may not exist a stone. The frog can jump on a stone, but it must not jump into the water.

Given a list of stones' positions (in units) in sorted ascending order, determine if the frog is able to cross the river by landing on the last stone. Initially, the frog is on the first stone and assume the first jump must be 1 unit.

If the frog's last jump was k units, then its next jump must be either k - 1, k, or k + 1 units. Note that the frog can only jump in the forward direction.

Note:

- The number of stones is ≥ 2 and is < 1,100.
- Each stone's position will be a non-negative integer < 2^{31}.
- The first stone's position is always 0.

Example 1:

```
[0,1,3,5,6,8,12,17]

There are a total of 8 stones.
The first stone at the 0th unit, second stone at the 1st unit,
third stone at the 3rd unit, and so on...
The last stone at the 17th unit.

Return true. The frog can jump to the last stone by jumping
1 unit to the 2nd stone, then 2 units to the 3rd stone, then
2 units to the 4th stone, then 3 units to the 6th stone,
4 units to the 7th stone, and 5 units to the 8th stone.
```

Example 2:

```
[0,1,2,3,4,8,9,11]

Return false. There is no way to jump to the last stone as
the gap between the 5th and 6th stone is too large.
```

Companies

Snapchat

Topics

Dynamic Programming

289. Game of Life (Medium)

According to the Wikipedia's article (https://en.wikipedia.org/wiki/Conway%27s_Game_of_Life): "The Game of Life , also known simply as Life , is a cellular automaton devised by the British mathematician John Horton Conway in 1970."

Given a board with m by n cells, each cell has an initial state live (1) or dead (0). Each cell interacts with its eight neighbors (https://en.wikipedia.org/wiki/Moore_neighborhood) (horizontal, vertical, diagonal) using the following four rules (taken from the above Wikipedia article):

1. Any live cell with fewer than two live neighbors dies, as if caused by under-population.
2. Any live cell with two or three live neighbors lives on to the next generation.
3. Any live cell with more than three live neighbors dies, as if by over-population..
4. Any dead cell with exactly three live neighbors becomes a live cell, as if by reproduction.

Write a function to compute the next state (after one update) of the board given its current state.

Follow up :

1. Could you solve it in-place? Remember that the board needs to be updated at the same time: You cannot update some cells first and then use their updated values to update other cells.
2. In this question, we represent the board using a 2D array. In principle, the board is infinite, which would cause problems when the active area encroaches the border of the array. How would you address these problems?

Companies

Google   Snapchat   Dropbox   Two Sigma

Topics

Array

134. Gas Station (Medium)

There are N gas stations along a circular route, where the amount of gas at station i is `gas[i]`.

You have a car with an unlimited gas tank and it costs `cost[i]` of gas to travel from station i to its next station (i+1). You begin the journey with an empty tank at one of the gas stations.

Return the starting gas station's index if you can travel around the circuit once, otherwise return -1.

Note:
The solution is guaranteed to be unique.

Companies

Topics

Greedy

320. Generalized Abbreviation (Medium)

Write a function to generate the generalized abbreviations of a word.

Example:

Given word = `word`, return the following list (order does not matter):

```
["word", "1ord", "w1rd", "wo1d", "wor1", "2rd", "w2d", "wo2", "1o1
d", "1or1", "w1r1", "1o2", "2r1", "3d", "w3", "4"]
```

Companies

Google

Topics

Backtracking   Bit Manipulation

22. Generate Parentheses (Medium)

Given n pairs of parentheses, write a function to generate all combinations of well-formed parentheses.

For example, given n = 3, a solution set is:

```
[
  "((()))",
  "(()())",
  "(())()",
  "()(())",
  "()()()"
]
```

Companies

Google   Uber   Zenefits

Topics

String   Backtracking

261. Graph Valid Tree (Medium)

Given `n` nodes labeled from `0` to `n - 1` and a list of undirected edges (each edge is a pair of nodes), write a function to check whether these edges make up a valid tree.

For example:

Given `n = 5` and `edges = [[0, 1], [0, 2], [0, 3], [1, 4]]`, return `true`.

Given `n = 5` and `edges = [[0, 1], [1, 2], [2, 3], [1, 3], [1, 4]]`, return `false`.

Note: you can assume that no duplicate edges will appear in `edges`. Since all edges are undirected, `[0, 1]` is the same as `[1, 0]` and thus will not appear together in `edges`.

Companies

Google Facebook Zenefits

Topics

Depth-first Search Breadth-first Search Union Find Graph

89. Gray Code (Medium)

The gray code is a binary numeral system where two successive values differ in only one bit.

Given a non-negative integer n representing the total number of bits in the code, print the sequence of gray code. A gray code sequence must begin with 0.

For example, given n = 2, return `[0,1,3,2]`. Its gray code sequence is:

```
00 - 0
01 - 1
11 - 3
10 - 2
```

Note:
For a given n, a gray code sequence is not uniquely defined.

For example, `[0,2,3,1]` is also a valid gray code sequence according to the above definition.

For now, the judge is able to judge based on one instance of gray code sequence. Sorry about that.

Companies

Amazon

Topics

Backtracking

49. Group Anagrams (Medium)

Given an array of strings, group anagrams together.

For example, given: `["eat", "tea", "tan", "ate", "nat", "bat"]`,
Return:

```
[
  ["ate", "eat","tea"],
  ["nat","tan"],
  ["bat"]
]
```

Note: All inputs will be in lower-case.

Companies

Facebook Amazon Bloomberg Uber Yelp

Topics

Hash Table String

249. Group Shifted Strings (Medium)

Given a string, we can "shift" each of its letter to its successive letter, for example:
`"abc" -> "bcd"`. We can keep "shifting" which forms the sequence:

```
"abc" -> "bcd" -> ... -> "xyz"
```

Given a list of strings which contains only lowercase alphabets, group all strings that belong to the same shifting sequence.

For example, given: `["abc", "bcd", "acef", "xyz", "az", "ba", "a", "z"]`,
A solution is:

```
[
  ["abc","bcd","xyz"],
  ["az","ba"],
  ["acef"],
  ["a","z"]
]
```

Companies

Google Uber

Topics

Hash Table String

375. Guess Number Higher or Lower II (Medium)

We are playing the Guess Game. The game is as follows:

I pick a number from 1 to n. You have to guess which number I picked.

Every time you guess wrong, I'll tell you whether the number I picked is higher or lower.

However, when you guess a particular number x, and you guess wrong, you pay $x. You win the game when you guess the number I picked.

Example:

```
n = 10, I pick 8.

First round:  You guess 5, I tell you that it's higher. You pay $5.
Second round: You guess 7, I tell you that it's higher. You pay $7.
Third round:  You guess 9, I tell you that it's lower. You pay $9.

Game over. 8 is the number I picked.

You end up paying $5 + $7 + $9 = $21.
```

Given a particular n ≥ 1, find out how much money you need to have to guarantee a win.

Companies

(Google)

Topics

(Dynamic Programming) (Minimax)

374. Guess Number Higher or Lower (Easy)

We are playing the Guess Game. The game is as follows:

I pick a number from 1 to n. You have to guess which number I picked.

Every time you guess wrong, I'll tell you whether the number is higher or lower.

You call a pre-defined API `guess(int num)` which returns 3 possible results (`-1`, `1`, or `0`):

```
-1 : My number is lower
 1 : My number is higher
 0 : Congrats! You got it!
```

Example:

```
n = 10, I pick 6.

Return 6.
```

Companies

Google

Topics

Binary Search

202. Happy Number (Easy)

Write an algorithm to determine if a number is "happy".

A happy number is a number defined by the following process: Starting with any positive integer, replace the number by the sum of the squares of its digits, and repeat the process until the number equals 1 (where it will stay), or it loops endlessly in a cycle which does not include 1. Those numbers for which this process ends in 1 are happy numbers.

Example: 19 is a happy number

$1^2 + 9^2 = 82$
$8^2 + 2^2 = 68$
$6^2 + 8^2 = 100$
$1^2 + 0^2 + 0^2 = 1$

Companies

Uber   Twitter   Airbnb

Topics

Hash Table   Math

275. H-Index II (Medium)

Follow up for H-Index (/problems/h-index/): What if the `citations` array is sorted in ascending order? Could you optimize your algorithm?

Companies

Facebook

Topics

Binary Search

274. H-Index (Medium)

Given an array of citations (each citation is a non-negative integer) of a researcher, write a function to compute the researcher's h-index.

According to the definition of h-index on Wikipedia (https://en.wikipedia.org/wiki/H-index): "A scientist has index h if h of his/her N papers have at least h citations each, and the other N − h papers have no more than   h citations each."

For example, given `citations = [3, 0, 6, 1, 5]`, which means the researcher has 5 papers in total and each of them had received `3, 0, 6, 1, 5` citations respectively. Since the researcher has `3` papers with at least `3` citations each and the remaining two with no more than   `3` citations each, his h-index is `3`.

Note: If there are several possible values for `h`, the maximum one is taken as the h-index.

Companies

Google   Facebook   Bloomberg

Topics

Hash Table   Sort

213. House Robber II (Medium)

Note: This is an extension of House Robber (https://leetcode.com/problems/house-robber/).

After robbing those houses on that street, the thief has found himself a new place for his thievery so that he will not get too much attention. This time, all houses at this place are arranged in a circle. That means the first house is the neighbor of the last one. Meanwhile, the security system for these houses remain the same as for those in the previous street.

Given a list of non-negative integers representing the amount of money of each house, determine the maximum amount of money you can rob tonight without alerting the police .

Companies

Microsoft

Topics

Dynamic Programming

337. House Robber III (Medium)

The thief has found himself a new place for his thievery again. There is only one entrance to this area, called the "root." Besides the root, each house has one and only one parent house. After a tour, the smart thief realized that "all houses in this place forms a binary tree". It will automatically contact the police if two directly-linked houses were broken into on the same night.

Determine the maximum amount of money the thief can rob tonight without alerting the police.

Example 1:

```
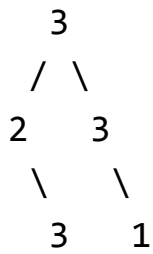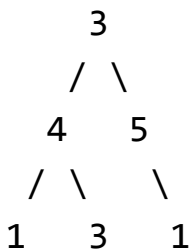    3
   / \
  2   3
   \   \
    3   1
```

Maximum amount of money the thief can rob = 3 + 3 + 1 = 7.

Example 2:

```
      3
     / \
    4   5
   / \   \
  1   3   1
```

Maximum amount of money the thief can rob = 4 + 5 = 9.

Companies

Uber

Topics

Tree   Depth-first Search

198. House Robber (Easy)

You are a professional robber planning to rob houses along a street. Each house has a certain amount of money stashed, the only constraint stopping you from robbing each of them is that adjacent houses have security system connected and it will automatically contact the police if two adjacent houses were broken into on the same night .

Given a list of non-negative integers representing the amount of money of each house, determine the maximum amount of money you can rob tonight without alerting the police .

Companies

LinkedIn   Airbnb

Topics

Dynamic Programming

232. Implement Queue using Stacks (Easy)

Implement the following operations of a queue using stacks.

- push(x) -- Push element x to the back of queue.
- pop() -- Removes the element from in front of queue.
- peek() -- Get the front element.
- empty() -- Return whether the queue is empty.

Notes:

- You must use only standard operations of a stack -- which means only `push to top`, `peek/pop from top`, `size`, and `is empty` operations are valid.
- Depending on your language, stack may not be supported natively. You may simulate a stack by using a list or deque (double-ended queue), as long as you use only standard operations of a stack.
- You may assume that all operations are valid (for example, no pop or peek operations will be called on an empty queue).

Companies

Microsoft Bloomberg

Topics

Stack Design

225. Implement Stack using Queues (Easy)

Implement the following operations of a stack using queues.

- push(x) -- Push element x onto stack.
- pop() -- Removes the element on top of the stack.
- top() -- Get the top element.
- empty() -- Return whether the stack is empty.

Notes:

- You must use only standard operations of a queue -- which means only `push to back`, `peek/pop from front`, `size`, and `is empty` operations are valid.
- Depending on your language, queue may not be supported natively. You may simulate a queue by using a list or deque (double-ended queue), as long as you use only standard operations of a queue.
- You may assume that all operations are valid (for example, no pop or top operations will be called on an empty stack).

Companies

Bloomberg

Topics

Stack   Design

28. Implement strStr() (Easy)

Implement strStr() (http://www.cplusplus.com/reference/cstring/strstr/).

Return the index of the first occurrence of needle in haystack, or -1 if needle is not part of haystack.

Example 1:

```
Input: haystack = "hello", needle = "ll"
Output: 2
```

Example 2:

```
Input: haystack = "aaaaa", needle = "bba"
Output: -1
```

Companies

Facebook   Microsoft   Apple   Pocket Gems

Topics

Two Pointers   String

208. Implement Trie (Prefix Tree) (Medium)

Implement a trie with `insert`, `search`, and `startsWith` methods.

Note:

You may assume that all inputs are consist of lowercase letters `a-z`.

Companies

Google Facebook Microsoft Bloomberg Uber Twitter

Topics

Design Trie

334. Increasing Triplet Subsequence (Medium)

Given an unsorted array return whether an increasing subsequence of length 3 exists or not in the array.

Formally the function should:

> Return true if there exists i, j, k
>
> such that arr[i] < arr[j] < arr[k] given 0 ≤ i < j < k ≤ n-1 else return false.

Your algorithm should run in O(n) time complexity and O(1) space complexity.

Examples:
Given `[1, 2, 3, 4, 5]`,
return `true`.

Given `[5, 4, 3, 2, 1]`,
return `false`.

Companies

Facebook

Topics

285. Inorder Successor in BST (Medium)

Given a binary search tree and a node in it, find the in-order successor of that node in the BST.

Note : If the given node has no in-order successor in the tree, return `null`.

Companies

Facebook   Microsoft   Pocket Gems

Topics

Tree

381. Insert Delete GetRandom O(1) - Duplicates allowed (Hard)

Design a data structure that supports all following operations in average O(1) time.

Note: Duplicate elements are allowed.

1. `insert(val)`: Inserts an item val to the collection.
2. `remove(val)`: Removes an item val from the collection if present.
3. `getRandom`: Returns a random element from current collection of elements. The probability of each element being returned is linearly related   to the number of same value the collection contains.

Example:

```
// Init an empty collection.
RandomizedCollection collection = new RandomizedCollection();

// Inserts 1 to the collection. Returns true as the collection did n
ot contain 1.
collection.insert(1);

// Inserts another 1 to the collection. Returns false as the collect
ion contained 1. Collection now contains [1,1].
collection.insert(1);

// Inserts 2 to the collection, returns true. Collection now contain
s [1,1,2].
collection.insert(2);

// getRandom should return 1 with the probability 2/3, and returns 2
with the probability 1/3.
collection.getRandom();

// Removes 1 from the collection, returns true. Collection now conta
ins [1,2].
collection.remove(1);

// getRandom should return 1 and 2 both equally likely.
collection.getRandom();
```

Companies

Yelp

Topics

Array   Hash Table   Design

380. Insert Delete GetRandom O(1) (Medium)

Design a data structure that supports all following operations in average O(1) time.

1. `insert(val)`: Inserts an item val to the set if not already present.
2. `remove(val)`: Removes an item val from the set if present.
3. `getRandom`: Returns a random element from current set of elements. Each element must have the same probability   of being returned.

Example:

```
// Init an empty set.
RandomizedSet randomSet = new RandomizedSet();

// Inserts 1 to the set. Returns true as 1 was inserted successfull
y.
randomSet.insert(1);

// Returns false as 2 does not exist in the set.
randomSet.remove(2);

// Inserts 2 to the set, returns true. Set now contains [1,2].
randomSet.insert(2);

// getRandom should return either 1 or 2 randomly.
randomSet.getRandom();

// Removes 1 from the set, returns true. Set now contains [2].
randomSet.remove(1);

// 2 was already in the set, so return false.
randomSet.insert(2);

// Since 2 is the only number in the set, getRandom always return 2.
randomSet.getRandom();
```

Companies

Google Facebook Amazon Uber Twitter Yelp Pocket Gems

Topics

Array Hash Table Design

57. Insert Interval (Hard)

Given a set of non-overlapping intervals, insert a new interval into the intervals (merge if necessary).

You may assume that the intervals were initially sorted according to their start times.

Example 1:
Given intervals `[1,3],[6,9]`, insert and merge `[2,5]` in as `[1,5],[6,9]`.

Example 2:
Given `[1,2],[3,5],[6,7],[8,10],[12,16]`, insert and merge `[4,9]` in as `[1,2]`, `[3,10],[12,16]`.

This is because the new interval `[4,9]` overlaps with `[3,5],[6,7],[8,10]`.

Companies

Google   Facebook   LinkedIn

Topics

Array   Sort

147. Insertion Sort List (Medium)

Sort a linked list using insertion sort.

Companies

Topics

Linked List   Sort

343. Integer Break (Medium)

Given a positive integer n, break it into the sum of at least two positive integers and maximize the product of those integers. Return the maximum product you can get.

For example, given n = 2, return 1 (2 = 1 + 1); given n = 10, return 36 (10 = 3 + 3 + 4).

Note: You may assume that n is not less than 2 and not larger than 58.

Companies

Topics

Math   Dynamic Programming

397. Integer Replacement (Medium)

Given a positive integer n and you can do operations as follow:

1. If n is even, replace n with `n/2`.
2. If n is odd, you can replace n with either `n + 1` or `n - 1`.

What is the minimum number of replacements needed for n to become 1?

Example 1:

```
Input:
8

Output:
3

Explanation:
8 -> 4 -> 2 -> 1
```

Example 2:

```
Input:
7

Output:
4

Explanation:
7 -> 8 -> 4 -> 2 -> 1
or
7 -> 6 -> 3 -> 2 -> 1
```

Companies

Google   Baidu

Topics

273. Integer to English Words (Hard)

Convert a non-negative integer to its english words representation. Given input is guaranteed to be less than 2^{31} - 1.

For example,

```
123 -> "One Hundred Twenty Three"
12345 -> "Twelve Thousand Three Hundred Forty Five"
1234567 -> "One Million Two Hundred Thirty Four Thousand Five Hundre
d Sixty Seven"
```

Companies

Facebook   Microsoft

Topics

Math   String

12. Integer to Roman (Medium)

Given an integer, convert it to a roman numeral.

Input is guaranteed to be within the range from 1 to 3999.

Companies

Twitter

Topics

Math String

97. Interleaving String (Hard)

Given s1, s2, s3, find whether s3 is formed by the interleaving of s1 and s2.

For example,

Given:

s1 = `"aabcc"`,

s2 = `"dbbca"`,

When s3 = `"aadbbcbcac"`, return true.

When s3 = `"aadbbbaccc"`, return false.

Companies

Topics

String   Dynamic Programming

350. Intersection of Two Arrays II (Easy)

Given two arrays, write a function to compute their intersection.

Example:
Given nums1 = `[1, 2, 2, 1]`, nums2 = `[2, 2]`, return `[2, 2]`.

Note:

- Each element in the result should appear as many times as it shows in both arrays.
- The result can be in any order.

Follow up:

- What if the given array is already sorted? How would you optimize your algorithm?
- What if nums1's size is small compared to nums2's size? Which algorithm is better?
- What if elements of nums2 are stored on disk, and the memory is limited such that you cannot load all elements into the memory at once?

Companies

Topics

Hash Table   Two Pointers   Binary Search   Sort

349. Intersection of Two Arrays (Easy)

Given two arrays, write a function to compute their intersection.

Example:
Given nums1 = `[1, 2, 2, 1]`, nums2 = `[2, 2]`, return `[2]`.

Note:

- Each element in the result must be unique.
- The result can be in any order.

Companies

Two Sigma

Topics

Hash Table · Two Pointers · Binary Search · Sort

160. Intersection of Two Linked Lists (Easy)

Write a program to find the node at which the intersection of two singly linked lists begins.

For example, the following two linked lists:

```
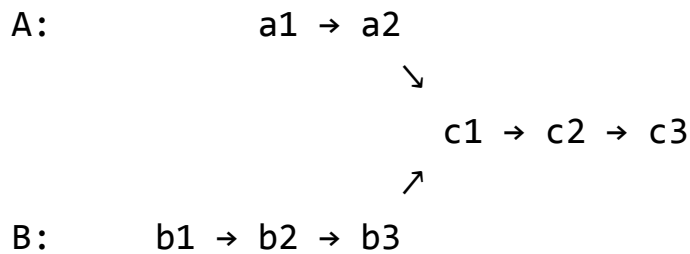A:              a1 → a2
                    ↘
                      c1 → c2 → c3
                    ↗
B:       b1 → b2 → b3
```

begin to intersect at node c1.

Notes:

- If the two linked lists have no intersection at all, return `null`.
- The linked lists must retain their original structure after the function returns.
- You may assume there are no cycles anywhere in the entire linked structure.
- Your code should preferably run in O(n) time and use only O(1) memory.

Companies

Microsoft   Amazon   Bloomberg   Airbnb

Topics

Linked List

226. Invert Binary Tree (Easy)

Invert a binary tree.

```
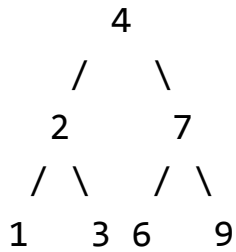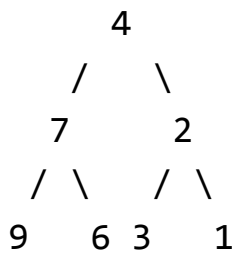     4
   /   \
  2     7
 / \   / \
1   3 6   9
```

to

```
     4
   /   \
  7     2
 / \   / \
9   6 3   1
```

Trivia:

This problem was inspired by this original tweet by Max Howell:

> Google: 90% of our engineers use the software you wrote (Homebrew), but you can't invert a binary tree on a whiteboard so fuck off.

Companies

Topics

Tree

392. Is Subsequence (Medium)

Given a string s and a string t, check if s is subsequence of t.

You may assume that there is only lower case English letters in both s and t. t is potentially a very long (length ~= 500,000) string, and s is a short string (<=100).

A subsequence of a string is a new string which is formed from the original string by deleting some (can be none) of the characters without disturbing the relative positions of the remaining characters. (ie, `"ace"` is a subsequence of `"abcde"` while `"aec"` is not).

Example 1:
s = `"abc"`, t = `"ahbgdc"`

Return `true`.

Example 2:
s = `"axc"`, t = `"ahbgdc"`

Return `false`.

Follow up:
If there are lots of incoming S, say S1, S2, ... , Sk where k >= 1B, and you want to check one by one to see if T has its subsequence. In this scenario, how would you change your code?

Companies

Pinterest

Topics

Binary Search   Dynamic Programming   Greedy

205. Isomorphic Strings (Easy)

Given two strings s and t, determine if they are isomorphic.

Two strings are isomorphic if the characters in s can be replaced to get t.

All occurrences of a character must be replaced with another character while preserving the order of characters. No two characters may map to the same character but a character may map to itself.

For example,
Given `"egg"`, `"add"`, return true.

Given `"foo"`, `"bar"`, return false.

Given `"paper"`, `"title"`, return true.

Note:
You may assume both s and t have the same length.

Companies

LinkedIn

Topics

Hash Table

45. Jump Game II (Hard)

Given an array of non-negative integers, you are initially positioned at the first index of the array.

Each element in the array represents your maximum jump length at that position.

Your goal is to reach the last index in the minimum number of jumps.

For example:
Given array A = `[2,3,1,1,4]`

The minimum number of jumps to reach the last index is `2`. (Jump `1` step from index 0 to 1, then `3` steps to the last index.)

Note:
You can assume that you can always reach the last index.

Companies

Topics

Array   Greedy

55. Jump Game (Medium)

Given an array of non-negative integers, you are initially positioned at the first index of the array.

Each element in the array represents your maximum jump length at that position.

Determine if you are able to reach the last index.

For example:
A = `[2,3,1,1,4]`, return `true`.

A = `[3,2,1,0,4]`, return `false`.

Companies

Microsoft

Topics

Array   Greedy

215. Kth Largest Element in an Array (Medium)

Find the kth largest element in an unsorted array. Note that it is the kth largest element in the sorted order, not the kth distinct element.

For example,
Given `[3,2,1,5,6,4]` and k = 2, return 5.

Note:
You may assume k is always valid, 1 ≤ k ≤ array's length.

Companies

Facebook   Microsoft   Amazon   Bloomberg   Apple   Pocket Gems

Topics

Divide and Conquer   Heap

230. Kth Smallest Element in a BST (Medium)

Given a binary search tree, write a function `kthSmallest` to find the kth smallest element in it.

Note:
You may assume k is always valid, 1 ≤ k ≤ BST's total elements.

Follow up:
What if the BST is modified (insert/delete operations) often and you need to find the kth smallest frequently? How would you optimize the kthSmallest routine?

Companies

Google Bloomberg Uber

Topics

Binary Search Tree

378. Kth Smallest Element in a Sorted Matrix (Medium)

Given a n x n matrix where each of the rows and columns are sorted in ascending order, find the kth smallest element in the matrix.

Note that it is the kth smallest element in the sorted order, not the kth distinct element.

Example:

```
matrix = [
    [ 1,  5,  9],
    [10, 11, 13],
    [12, 13, 15]
],
k = 8,

return 13.
```

Note:
You may assume k is always valid, $1 \leq k \leq n^2$.

Companies

Google   Twitter

Topics

Binary Search   Heap

333. Largest BST Subtree (Medium)

Given a binary tree, find the largest subtree which is a Binary Search Tree (BST), where largest means subtree with largest number of nodes in it.

Note:
A subtree must include all of its descendants.
Here's an example:

```
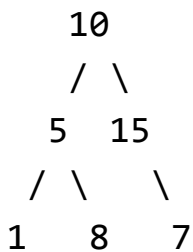    10
    / \
   5   15
  / \   \
 1   8   7
```

The Largest BST Subtree in this case is the highlighted one.
The return value is the subtree's size, which is 3.

Follow up:
Can you figure out ways to solve it with O(n) time complexity?

Companies

Microsoft

Topics

Tree

368. Largest Divisible Subset (Medium)

Given a set of distinct positive integers, find the largest subset such that every pair (S_i, S_j) of elements in this subset satisfies: $S_i \% S_j = 0$ or $S_j \% S_i = 0$.

If there are multiple solutions, return any subset is fine.

Example 1:

```
nums: [1,2,3]

Result: [1,2] (of course, [1,3] will also be ok)
```

Example 2:

```
nums: [1,2,4,8]

Result: [1,2,4,8]
```

Companies

Google

Topics

Math   Dynamic Programming

179. Largest Number (Medium)

Given a list of non negative integers, arrange them such that they form the largest number.

For example, given `[3, 30, 34, 5, 9]`, the largest formed number is `9534330`.

Note: The result may be very large, so you need to return a string instead of an integer.

Companies

Works Applications

Topics

Sort

84. Largest Rectangle in Histogram (Hard)

Given n non-negative integers representing the histogram's bar height where the width of each bar is 1, find the area of largest rectangle in the histogram.

Above is a histogram where width of each bar is 1, given height = `[2,1,5,6,2,3]` .

The largest rectangle is shown in the shaded area, which has area = `10` unit.

For example,
Given heights = `[2,1,5,6,2,3]`,
return `10`.

Companies

Topics

(Array) (Stack)

58. Length of Last Word (Easy)

Given a string s consists of upper/lower-case alphabets and empty space characters ' ', return the length of last word in the string.

If the last word does not exist, return 0.

Note: A word is defined as a character sequence consists of non-space characters only.

Example:

```
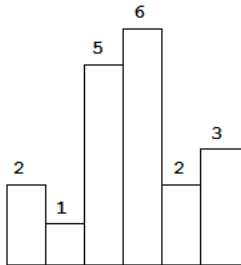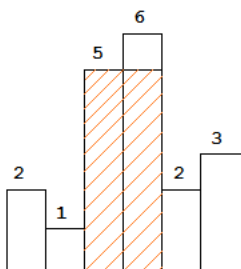Input: "Hello World"
Output: 5
```

Companies

Topics

String

17. Letter Combinations of a Phone Number (Medium)

Given a digit string, return all possible letter combinations that the number could represent.

A mapping of digit to letters (just like on the telephone buttons) is given below.


```
Input:Digit string "23"
Output: ["ad", "ae", "af", "bd", "be", "bf", "cd", "ce", "cf"].
```

Note:
Although the above answer is in lexicographical order, your answer could be in any order you want.

Companies

Google   Facebook   Amazon   Uber   Dropbox

Topics

String   Backtracking

386. Lexicographical Numbers (Medium)

Given an integer n, return 1 - n in lexicographical order.

For example, given 13, return: [1,10,11,12,13,2,3,4,5,6,7,8,9].

Please optimize your algorithm to use less time and space. The input size may be as large as 5,000,000.

Companies

Bloomberg

Topics

356. Line Reflection (Medium)

Given n points on a 2D plane, find if there is such a line parallel to y-axis that reflect the given points.

Example 1:

Given points = `[[1,1],[-1,1]]`, return `true`.

Example 2:

Given points = `[[1,1],[-1,-1]]`, return `false`.

Follow up:
Could you do better than O(n^2)?

Companies

Google

Topics

Hash Table   Math

142. Linked List Cycle II (Medium)

Given a linked list, return the node where the cycle begins. If there is no cycle, return `null`.

Note: Do not modify the linked list.

Follow up：
Can you solve it without using extra space?

Companies

Topics

Linked List   Two Pointers

141. Linked List Cycle (Easy)

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

Follow up:
Can you solve it without using extra space?

Companies

Microsoft   Amazon   Bloomberg   Yahoo

Topics

Linked List   Two Pointers

382. Linked List Random Node (Medium)

Given a singly linked list, return a random node's value from the linked list. Each node must have the same probability   of being chosen.

Follow up:
What if the linked list is extremely large and its length is unknown to you? Could you solve this efficiently without using extra space?

Example:

```
// Init a singly linked list [1,2,3].
ListNode head = new ListNode(1);
head.next = new ListNode(2);
head.next.next = new ListNode(3);
Solution solution = new Solution(head);

// getRandom() should return either 1, 2, or 3 randomly. Each elemen
t should have equal probability of returning.
solution.getRandom();
```

Companies

Google

Topics

Reservoir Sampling

359. Logger Rate Limiter (Easy)

Design a logger system that receive stream of messages along with its timestamps, each message should be printed if and only if it is not printed in the last 10 seconds .

Given a message and a timestamp (in seconds granularity), return true if the message should be printed in the given timestamp, otherwise returns false.

It is possible that several messages arrive roughly at the same time.

Example:

```
Logger logger = new Logger();

// logging string "foo" at timestamp 1
logger.shouldPrintMessage(1, "foo"); returns true;

// logging string "bar" at timestamp 2
logger.shouldPrintMessage(2,"bar"); returns true;

// logging string "foo" at timestamp 3
logger.shouldPrintMessage(3,"foo"); returns false;

// logging string "bar" at timestamp 8
logger.shouldPrintMessage(8,"bar"); returns false;

// logging string "foo" at timestamp 10
logger.shouldPrintMessage(10,"foo"); returns false;

// logging string "foo" at timestamp 11
logger.shouldPrintMessage(11,"foo"); returns true;
```

Companies

Google

Topics

Hash Table   Design

388. Longest Absolute File Path (Medium)

Suppose we abstract our file system by a string in the following manner:

The string `"dir\n\tsubdir1\n\tsubdir2\n\t\tfile.ext"` represents:

```
dir
    subdir1
    subdir2
        file.ext
```

The directory `dir` contains an empty sub-directory `subdir1` and a sub-directory `subdir2` containing a file `file.ext`.

The string
`"dir\n\tsubdir1\n\t\tfile1.ext\n\t\tsubsubdir1\n\tsubdir2\n\t\tsubsubdir2\n\t\t`
represents:

```
dir
    subdir1
        file1.ext
        subsubdir1
    subdir2
        subsubdir2
            file2.ext
```

The directory `dir` contains two sub-directories `subdir1` and `subdir2`. `subdir1` contains a file `file1.ext` and an empty second-level sub-directory `subsubdir1`. `subdir2` contains a second-level sub-directory `subsubdir2` containing a file `file2.ext`.

We are interested in finding the longest (number of characters) absolute path to a file within our file system. For example, in the second example above, the longest absolute path is `"dir/subdir2/subsubdir2/file2.ext"`, and its length is `32` (not including the double quotes).

Given a string representing the file system in the above format, return the length of the longest absolute path to file in the abstracted file system. If there is no file in the system, return `0`.

Note:

- The name of a file contains at least a `.` and an extension.
- The name of a directory or sub-directory will not contain a `.`.

Time complexity required: `O(n)` where `n` is the size of the input string.

Notice that `a/aa/aaa/file1.txt` is not the longest file path, if there is another path `aaaaaaaaaaaaaaaaaaaaa/sth.png`.

Companies

Google

Topics

14. Longest Common Prefix (Easy)

Write a function to find the longest common prefix string amongst an array of strings.

Companies

Yelp

Topics

String

128. Longest Consecutive Sequence (Hard)

Given an unsorted array of integers, find the length of the longest consecutive elements sequence.

For example,
Given `[100, 4, 200, 1, 3, 2]`,
The longest consecutive elements sequence is `[1, 2, 3, 4]`. Return its length: `4`.

Your algorithm should run in O(n) complexity.

Companies

Google   Facebook

Topics

Array   Union Find

329. Longest Increasing Path in a Matrix (Hard)

Given an integer matrix, find the length of the longest increasing path.

From each cell, you can either move to four directions: left, right, up or down. You may NOT move diagonally or move outside of the boundary (i.e. wrap-around is not allowed).

Example 1:

```
nums = [
  [9,9,4],
  [6,6,8],
  [2,1,1]
]
```

Return 4
The longest increasing path is [1, 2, 6, 9].

Example 2:

```
nums = [
  [3,4,5],
  [3,2,6],
  [2,2,1]
]
```

Return 4
The longest increasing path is [3, 4, 5, 6]. Moving diagonally is not allowed.

Companies

Google

Topics

Depth-first Search   Topological Sort   Memoization

300. Longest Increasing Subsequence (Medium)

Given an unsorted array of integers, find the length of longest increasing subsequence.

For example,
Given `[10, 9, 2, 5, 3, 7, 101, 18]`,
The longest increasing subsequence is `[2, 3, 7, 101]`, therefore the length is `4`.
Note that there may be more than one LIS combination, it is only necessary for you to return the length.

Your algorithm should run in O(n^2) complexity.

Follow up: Could you improve it to O(n log n) time complexity?

Companies

Microsoft

Topics

Binary Search   Dynamic Programming

409. Longest Palindrome (Easy)

Given a string which consists of lowercase or uppercase letters, find the length of the longest palindromes that can be built with those letters.

This is case sensitive, for example `"Aa"` is not considered a palindrome here.

Note:
Assume the length of given string will not exceed 1,010.

Example:

```
Input:
"abccccdd"

Output:
7

Explanation:
One longest palindrome that can be built is "dccaccd", whose length
 is 7.
```

Companies

Google

Topics

Hash Table

5. Longest Palindromic Substring (Medium)

Given a string s, find the longest palindromic substring in s. You may assume that the maximum length of s is 1000.

Example:

```
Input: "babad"

Output: "bab"

Note: "aba" is also a valid answer.
```

Example:

```
Input: "cbbd"

Output: "bb"
```

Companies

Microsoft   Amazon   Bloomberg

Topics

String

395. Longest Substring with At Least K Repeating Characters (Medium)

Find the length of the longest substring T of a given string (consists of lowercase letters only) such that every character in T appears no less than k times.

Example 1:

```
Input:
s = "aaabb", k = 3

Output:
3

The longest substring is "aaa", as 'a' is repeated 3 times.
```

Example 2:

```
Input:
s = "ababbc", k = 2

Output:
5

The longest substring is "ababb", as 'a' is repeated 2 times and 'b'
is repeated 3 times.
```

Companies

Baidu

Topics

340. Longest Substring with At Most K Distinct Characters (Hard)

Given a string, find the length of the longest substring T that contains at most k distinct characters.

For example, Given s = "eceba" and k = 2,

T is "ece" which its length is 3.

Companies

Google AppDynamics Coupang

Topics

Hash Table String

159. Longest Substring with At Most Two Distinct Characters (Hard)

Given a string, find the length of the longest substring T that contains at most 2 distinct characters.

For example, Given s = "eceba",

T is "ece" which its length is 3.

Companies

Google

Topics

Hash Table   Two Pointers   String

3. Longest Substring Without Repeating Characters (Medium)

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

Examples:

Given `"abcabcbb"`, the answer is `"abc"`, which the length is 3.

Given `"bbbbb"`, the answer is `"b"`, with the length of 1.

Given `"pwwkew"`, the answer is `"wke"`, with the length of 3. Note that the answer must be a substring , `"pwke"` is a subsequence and not a substring.

Companies

Amazon   Bloomberg   Yelp   Adobe

Topics

Hash Table   Two Pointers   String

32. Longest Valid Parentheses (Hard)

Given a string containing just the characters `'('` and `')'`, find the length of the longest valid (well-formed) parentheses substring.

For `"(()"`, the longest valid parentheses substring is `"()"`, which has length = 2.

Another example is `")()())"`, where the longest valid parentheses substring is `"()()"`, which has length = 4.

Companies

Topics

String   Dynamic Programming

235. Lowest Common Ancestor of a Binary Search Tree (Easy)

Given a binary search tree (BST), find the lowest common ancestor (LCA) of two given nodes in the BST.

According to the definition of LCA on Wikipedia (https://en.wikipedia.org/wiki/Lowest_common_ancestor): "The lowest common ancestor is defined between two nodes v and w as the lowest node in T that has both v and w as descendants (where we allow a node to be a descendant of itself)."

```
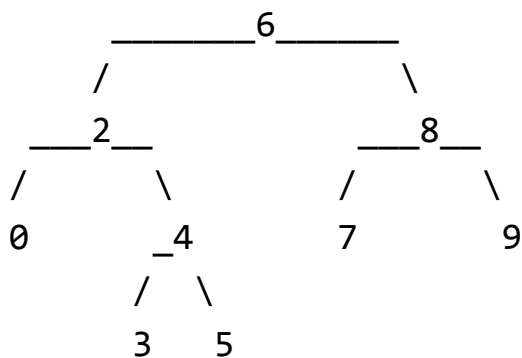        _____6_____
       /                 \
    ___2__             ___8__
   /      \           /      \
  0       _4         7        9
         /  \
        3    5
```

For example, the lowest common ancestor (LCA) of nodes `2` and `8` is `6`. Another example is LCA of nodes `2` and `4` is `2`, since a node can be a descendant of itself according to the LCA definition.

Companies

Facebook · Microsoft · Amazon · Twitter

Topics

Tree

236. Lowest Common Ancestor of a Binary Tree (Medium)

Given a binary tree, find the lowest common ancestor (LCA) of two given nodes in the tree.

According to the definition of LCA on Wikipedia (https://en.wikipedia.org/wiki/Lowest_common_ancestor): "The lowest common ancestor is defined between two nodes v and w as the lowest node in T that has both v and w as descendants (where we allow a node to be a descendant of itself)."

```
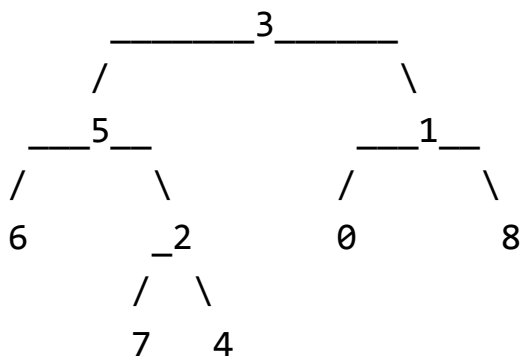        _____3_____
       /                 \
    ___5__            ___1__
   /      \          /      \
  6       _2        0        8
         /  \
        7    4
```

For example, the lowest common ancestor (LCA) of nodes 5 and 1 is 3. Another example is LCA of nodes 5 and 4 is 5, since a node can be a descendant of itself according to the LCA definition.

Companies

Facebook   Microsoft   Amazon   LinkedIn   Apple

Topics

Tree

146. LRU Cache (Hard)

Design and implement a data structure for Least Recently Used (LRU) cache (https://en.wikipedia.org/wiki/Cache_replacement_policies#LRU). It should support the following operations: `get` and `put`.

`get(key)` - Get the value (will always be positive) of the key if the key exists in the cache, otherwise return -1.

`put(key, value)` - Set or insert the value if the key is not already present. When the cache reached its capacity, it should invalidate the least recently used item before inserting a new item.

Follow up:
Could you do both operations in O(1) time complexity?

Example:

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

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 (not found)
cache.put(4, 4);    // evicts key 1
cache.get(1);       // returns -1 (not found)
cache.get(3);       // returns 3
cache.get(4);       // returns 4
```

Companies

Google   Facebook   Microsoft   Amazon   Bloomberg   Uber   Twitter   Snapchat   Zenefits   Yahoo
Palantir

Topics

Design

229. Majority Element II (Medium)

Given an integer array of size n, find all elements that appear more than $\lfloor n/3 \rfloor$ times. The algorithm should run in linear time and in O(1) space.

Companies

Zenefits

Topics

Array

169. Majority Element (Easy)

Given an array of size n, find the majority element. The majority element is the element that appears more than `⌊ n/2 ⌋` times.

You may assume that the array is non-empty and the majority element always exist in the array.

Companies

Zenefits   Adobe

Topics

Array   Divide and Conquer   Bit Manipulation

149. Max Points on a Line (Hard)

Given n points on a 2D plane, find the maximum number of points that lie on the same straight line.

Companies

LinkedIn   Twitter   Apple

Topics

Hash Table   Math

363. Max Sum of Rectangle No Larger Than K (Hard)

Given a non-empty 2D matrix matrix and an integer k, find the max sum of a rectangle in the matrix such that its sum is no larger than k.

Example:

```
Given matrix = [
   [1,  0, 1],
   [0, -2, 3]
]
k = 2
```

The answer is 2. Because the sum of rectangle [[0, 1], [-2, 3]] is 2 and 2 is the max number no larger than k (k = 2).

Note:

1. The rectangle inside the matrix must have an area > 0.
2. What if the number of rows is much larger than the number of columns?

Companies

Google

Topics

Binary Search   Dynamic Programming   Queue

85. Maximal Rectangle (Hard)

Given a 2D binary matrix filled with 0's and 1's, find the largest rectangle containing only 1's and return its area.

For example, given the following matrix:

```
1 0 1 0 0
1 0 1 1 1
1 1 1 1 1
1 0 0 1 0
```

Return 6.

Companies

Facebook

Topics

Array   Hash Table   Dynamic Programming   Stack

221. Maximal Square (Medium)

Given a 2D binary matrix filled with 0's and 1's, find the largest square containing only 1's and return its area.

For example, given the following matrix:

```
1 0 1 0 0
1 0 1 1 1
1 1 1 1 1
1 0 0 1 0
```

Return 4.

Companies

Facebook   Apple   Airbnb

Topics

Dynamic Programming

104. Maximum Depth of Binary Tree (Easy)

Given a binary tree, find its maximum depth.

The maximum depth is the number of nodes along the longest path from the root node down to the farthest leaf node.

Companies

Uber · LinkedIn · Apple · Yahoo

Topics

Tree · Depth-first Search

164. Maximum Gap (Hard)

Given an unsorted array, find the maximum difference between the successive elements in its sorted form.

Try to solve it in linear time/space.

Return 0 if the array contains less than 2 elements.

You may assume all elements in the array are non-negative integers and fit in the 32-bit signed integer range.

Companies

Topics

(Sort)

318. Maximum Product of Word Lengths (Medium)

Given a string array `words`, find the maximum value of `length(word[i]) * length(word[j])` where the two words do not share common letters. You may assume that each word will contain only lower case letters. If no such two words exist, return 0.

Example 1:

Given `["abcw", "baz", "foo", "bar", "xtfn", "abcdef"]`
Return `16`
The two words can be `"abcw", "xtfn"`.

Example 2:

Given `["a", "ab", "abc", "d", "cd", "bcd", "abcd"]`
Return `4`
The two words can be `"ab", "cd"`.

Example 3:

Given `["a", "aa", "aaa", "aaaa"]`
Return `0`
No such pair of words.

Companies

Google

Topics

Bit Manipulation

152. Maximum Product Subarray (Medium)

Find the contiguous subarray within an array (containing at least one number) which has the largest product.

For example, given the array `[2,3,-2,4]`,
the contiguous subarray `[2,3]` has the largest product = `6`.

Companies

LinkedIn

Topics

Array   Dynamic Programming

325. Maximum Size Subarray Sum Equals k (Medium)

Given an array nums and a target value k, find the maximum length of a subarray that sums to k. If there isn't one, return 0 instead.

Note:
The sum of the entire nums array is guaranteed to fit within the 32-bit signed integer range.

Example 1:

Given nums = `[1, -1, 5, -2, 3]`, k = `3`,
return `4`. (because the subarray `[1, -1, 5, -2]` sums to 3 and is the longest)

Example 2:

Given nums = `[-2, -1, 2, 1]`, k = `1`,
return `2`. (because the subarray `[-1, 2]` sums to 1 and is the longest)

Follow Up:
Can you do it in O(n) time?

Companies
Facebook   Palantir

Topics
Hash Table

53. Maximum Subarray (Easy)

Find the contiguous subarray within an array (containing at least one number) which has the largest sum.

For example, given the array `[-2,1,-3,4,-1,2,1,-5,4]`,
the contiguous subarray `[4,-1,2,1]` has the largest sum = `6`.

More practice:

If you have figured out the O(n) solution, try coding another solution using the divide and conquer approach, which is more subtle.

Companies

Microsoft   Bloomberg   LinkedIn

Topics

Array   Divide and Conquer   Dynamic Programming

4. Median of Two Sorted Arrays (Hard)

There are two sorted arrays nums1 and nums2 of size m and n respectively.

Find the median of the two sorted arrays. The overall run time complexity should be O(log (m+n)).

Example 1:

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

The median is 2.0
```

Example 2:

```
nums1 = [1, 2]
nums2 = [3, 4]

The median is (2 + 3)/2 = 2.5
```

Companies

Google   Microsoft   Apple   Zenefits   Yahoo   Adobe   Dropbox

Topics

Array   Binary Search   Divide and Conquer

253. Meeting Rooms II (Medium)

Given an array of meeting time intervals consisting of start and end times `[[s1,e1],
[s2,e2],...]` (s_i < e_i), find the minimum number of conference rooms required.

For example,
Given `[[0, 30],[5, 10],[15, 20]]`,
return `2`.

Companies

Google Facebook Snapchat

Topics

Heap Greedy Sort

252. Meeting Rooms (Easy)

Given an array of meeting time intervals consisting of start and end times `[[s1,e1],` `[s2,e2],...]` ($s_i < e_i$), determine if a person could attend all meetings.

For example,
Given `[[0, 30],[5, 10],[15, 20]]`,
return `false`.

Companies

Facebook

Topics

Sort

56. Merge Intervals (Medium)

Given a collection of intervals, merge all overlapping intervals.

For example,
Given `[1,3],[2,6],[8,10],[15,18]`,
return `[1,6],[8,10],[15,18]`.

Companies

Google   Facebook   Microsoft   Bloomberg   LinkedIn   Twitter   Yelp

Topics

Array   Sort

23. Merge k Sorted Lists (Hard)

Merge k sorted linked lists and return it as one sorted list. Analyze and describe its complexity.

Companies

Google   Facebook   Microsoft   Amazon   Uber   LinkedIn   Twitter   Airbnb   IXL

Topics

Linked List   Divide and Conquer   Heap

88. Merge Sorted Array (Easy)

Given two sorted integer arrays nums1 and nums2, merge nums2 into nums1 as one sorted array.

Note:
You may assume that nums1 has enough space (size that is greater or equal to m + n) to hold additional elements from nums2. The number of elements initialized in nums1 and nums2 are m and n respectively.

Companies

Facebook   Microsoft   Bloomberg

Topics

Array   Two Pointers

21. Merge Two Sorted Lists (Easy)

Merge two sorted linked lists and return it as a new list. The new list should be made by splicing together the nodes of the first two lists.

Companies

Microsoft   Amazon   LinkedIn   Apple

Topics

Linked List

155. Min Stack (Easy)

Design a stack that supports push, pop, top, and retrieving the minimum element in constant time.

- push(x) -- Push element x onto stack.
- pop() -- Removes the element on top of the stack.
- top() -- Get the top element.
- getMin() -- Retrieve the minimum element in the stack.

Example:

```
MinStack minStack = new MinStack();
minStack.push(-2);
minStack.push(0);
minStack.push(-3);
minStack.getMin();   --> Returns -3.
minStack.pop();
minStack.top();      --> Returns 0.
minStack.getMin();   --> Returns -2.
```

Companies

Google Amazon Bloomberg Uber Snapchat Zenefits

Topics

Stack Design

385. Mini Parser (Medium)

Given a nested list of integers represented as a string, implement a parser to deserialize it.

Each element is either an integer, or a list -- whose elements may also be integers or other lists.

Note: You may assume that the string is well-formed:

- String is non-empty.
- String does not contain white spaces.
- String contains only digits `0-9`, `[`, `-` `,`, `]`.

Example 1:

```
Given s = "324",

You should return a NestedInteger object which contains a single int
eger 324.
```

Example 2:

```
Given s = "[123,[456,[789]]]",

Return a NestedInteger object containing a nested list with 2 elemen
ts:

1. An integer containing value 123.
2. A nested list containing two elements:
    i.  An integer containing value 456.
    ii. A nested list with one element:
         a. An integer containing value 789.
```

Companies

Airbnb

Topics

String Stack

111. Minimum Depth of Binary Tree (Easy)

Given a binary tree, find its minimum depth.

The minimum depth is the number of nodes along the shortest path from the root node down to the nearest leaf node.

Companies

Topics

Tree   Depth-first Search   Breadth-first Search

310. Minimum Height Trees (Medium)

For a undirected graph with tree characteristics, we can choose any node as the root. The result graph is then a rooted tree. Among all possible rooted trees, those with minimum height are called minimum height trees (MHTs). Given such a graph, write a function to find all the MHTs and return a list of their root labels.

Format
The graph contains `n` nodes which are labeled from `0` to `n - 1`. You will be given the number `n` and a list of undirected `edges` (each edge is a pair of labels).

You can assume that no duplicate edges will appear in `edges`. Since all edges are undirected, `[0, 1]` is the same as `[1, 0]` and thus will not appear together in `edges`.

Example 1:

Given `n = 4`, edges = `[[1, 0], [1, 2], [1, 3]]`

```
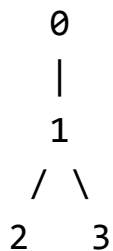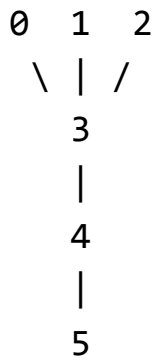        0
        |
        1
       / \
      2   3
```

return `[1]`

Example 2:

Given `n = 6`, edges = `[[0, 3], [1, 3], [2, 3], [4, 3], [5, 4]]`

```
     0   1   2
      \  |  /
         3
         |
         4
         |
         5
```

return `[3, 4]`

Note:

(1) According to the definition of tree on Wikipedia (https://en.wikipedia.org/wiki/Tree_(graph_theory)): "a tree is an undirected graph in which any two vertices are connected by exactly one path. In other words, any connected graph without simple cycles is a tree."

(2) The height of a rooted tree is the number of edges on the longest downward path between the root and a leaf.

Companies

(Google)

Topics

(Breadth-first Search) (Graph)

64. Minimum Path Sum (Medium)

Given a m x n grid filled with non-negative numbers, find a path from top left to bottom right which minimizes the sum of all numbers along its path.

Note: You can only move either down or right at any point in time.

Example 1:

```
[[1,3,1],
 [1,5,1],
 [4,2,1]]
```

Given the above grid map, return 7. Because the path 1→3→1→1→1 minimizes the sum.

Companies

Topics

Array   Dynamic Programming

209. Minimum Size Subarray Sum (Medium)

Given an array of n positive integers and a positive integer s, find the minimal length of a contiguous   subarray of which the sum ≥ s. If there isn't one, return 0 instead.

For example, given the array `[2,3,1,2,4,3]` and `s = 7`,
the subarray `[4,3]` has the minimal length under the problem constraint.

More practice:

If you have figured out the O(n) solution, try coding another solution of which the time complexity is O(n log n).

Companies

Facebook

Topics

Array   Two Pointers   Binary Search

411. Minimum Unique Word Abbreviation (Hard)

A string such as `"word"` contains the following abbreviations:

```
["word", "1ord", "w1rd", "wo1d", "wor1", "2rd", "w2d", "wo2", "1o1
d", "1or1", "w1r1", "1o2", "2r1", "3d", "w3", "4"]
```

Given a target string and a set of strings in a dictionary, find an abbreviation of this target string with the smallest possible   length such that it does not conflict with abbreviations of the strings in the dictionary.

Each number or letter in the abbreviation is considered length = 1. For example, the abbreviation "a32bc" has length = 4.

Note:

- In the case of multiple answers as shown in the second example below, you may return any one of them.
- Assume length of target string = m, and dictionary size = n. You may assume that $m \leq 21$, $n \leq 1000$, and $\log_2(n) + m \leq 20$.

Examples:

```
"apple", ["blade"] -> "a4" (because "5" or "4e" conflicts with "blad
e")

"apple", ["plain", "amber", "blade"] -> "1p3" (other valid answers i
nclude "ap3", "a3e", "2p2", "3le", "3l1").
```

Companies

Google

Topics

Backtracking   Bit Manipulation

76. Minimum Window Substring (Hard)

Given a string S and a string T, find the minimum window in S which will contain all the characters in T in complexity O(n).

For example,
S = `"ADOBECODEBANC"`
T = `"ABC"`

Minimum window is `"BANC"`.

Note:
If there is no such window in S that covers all characters in T, return the empty string `""`.

If there are multiple such windows, you are guaranteed that there will always be only one unique minimum window in S.

Companies

Facebook   Uber   LinkedIn   Snapchat

Topics

Hash Table   Two Pointers   String

268. Missing Number (Easy)

Given an array containing n distinct numbers taken from `0, 1, 2, ..., n`, find the one that is missing from the array.

For example,
Given nums = `[0, 1, 3]` return `2`.

Note:
Your algorithm should run in linear runtime complexity. Could you implement it using only constant extra space complexity?

Companies

Microsoft   Bloomberg

Topics

Array   Math   Bit Manipulation

163. Missing Ranges (Medium)

Given a sorted integer array where the range of elements are in the inclusive range [lower , upper], return its missing ranges.

For example, given `[0, 1, 3, 50, 75]`, lower = 0 and upper = 99, return `["2", "4->49", "51->74", "76->99"]`.

Companies

Google

Topics

Array

283. Move Zeroes (Easy)

Given an array `nums`, write a function to move all `0`'s to the end of it while maintaining the relative order of the non-zero elements.

For example, given `nums = [0, 1, 0, 3, 12]`, after calling your function, `nums` should be `[1, 3, 12, 0, 0]`.

Note:

1. You must do this in-place without making a copy of the array.
2. Minimize the total number of operations.

Companies

(Facebook) (Bloomberg)

Topics

(Array) (Two Pointers)

346. Moving Average from Data Stream (Easy)

Given a stream of integers and a window size, calculate the moving average of all integers in the sliding window.

For example,

```
MovingAverage m = new MovingAverage(3);
m.next(1) = 1
m.next(10) = (1 + 10) / 2
m.next(3) = (1 + 10 + 3) / 3
m.next(5) = (10 + 3 + 5) / 3
```

Companies

Google

Topics

Design   Queue

43. Multiply Strings (Medium)

Given two non-negative integers `num1` and `num2` represented as strings, return the product of `num1` and `num2`.

Note:

1. The length of both `num1` and `num2` is < 110.
2. Both `num1` and `num2` contains only digits `0-9`.
3. Both `num1` and `num2` does not contain any leading zero.
4. You must not use any built-in BigInteger library   or convert the inputs to integer directly.

Companies

(Facebook) (Twitter)

Topics

(Math) (String)

364. Nested List Weight Sum II (Medium)

Given a nested list of integers, return the sum of all integers in the list weighted by their depth.

Each element is either an integer, or a list -- whose elements may also be integers or other lists.

Different from the previous question (https://leetcode.com/problems/nested-list-weight-sum/) where weight is increasing from root to leaf, now the weight is defined from bottom up. i.e., the leaf level integers have weight 1, and the root level integers have the largest weight.

Example 1:
Given the list `[[1,1],2,[1,1]]`, return 8. (four 1's at depth 1, one 2 at depth 2)

Example 2:
Given the list `[1,[4,[6]]]`, return 17. (one 1 at depth 3, one 4 at depth 2, and one 6 at depth 1; 1*3 + 4*2 + 6*1 = 17)

Companies

(LinkedIn)

Topics

(Depth-first Search)

339. Nested List Weight Sum (Easy)

Given a nested list of integers, return the sum of all integers in the list weighted by their depth.

Each element is either an integer, or a list -- whose elements may also be integers or other lists.

Example 1:
Given the list `[[1,1],2,[1,1]]`, return 10. (four 1's at depth 2, one 2 at depth 1)

Example 2:
Given the list `[1,[4,[6]]]`, return 27. (one 1 at depth 1, one 4 at depth 2, and one 6 at depth 3; 1 + 4*2 + 6*3 = 27)

Companies

LinkedIn

Topics

Depth-first Search

31. Next Permutation (Medium)

Implement next permutation, which rearranges numbers into the lexicographically next greater permutation of numbers.

If such arrangement is not possible, it must rearrange it as the lowest possible order (ie, sorted in ascending order).

The replacement must be in-place, do not allocate extra memory.

Here are some examples. Inputs are in the left-hand column and its corresponding outputs are in the right-hand column.

```
1,2,3 → 1,3,2
3,2,1 → 1,2,3
1,1,5 → 1,5,1
```

Companies

Google

Topics

Array

292. Nim Game (Easy)

You are playing the following Nim Game with your friend: There is a heap of stones on the table, each time one of you take turns to remove 1 to 3 stones. The one who removes the last stone will be the winner. You will take the first turn to remove the stones.

Both of you are very clever and have optimal strategies for the game. Write a function to determine whether you can win the game given the number of stones in the heap.

For example, if there are 4 stones in the heap, then you will never win the game: no matter 1, 2, or 3 stones you remove, the last stone will always be removed by your friend.

Companies

Adobe

Topics

Brainteaser

52. N-Queens II (Hard)

Follow up for N-Queens problem.

Now, instead outputting board configurations, return the total number of distinct solutions.

One solution to the eight queens puzzle

Companies

Zenefits

Topics

Backtracking

51. N-Queens (Hard)

The n-queens puzzle is the problem of placing n queens on an n×n chessboard such that no two queens attack each other.

One solution to the eight queens puzzle

Given an integer n, return all distinct solutions to the n-queens puzzle.

Each solution contains a distinct board configuration of the n-queens' placement, where `'Q'` and `'.'` both indicate a queen and an empty space respectively.

For example,
There exist two distinct solutions to the 4-queens puzzle:

```
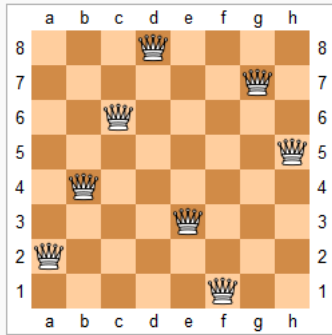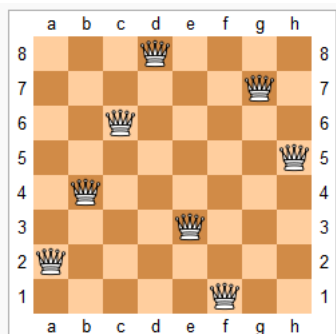[
 [".Q..",  // Solution 1
  "...Q",
  "Q...",
  "..Q."],

 ["..Q.",  // Solution 2
  "Q...",
  "...Q",
  ".Q.."]
]
```

Companies

Topics

Backtracking

400. Nth Digit (Easy)

Find the nth digit of the infinite integer sequence 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, ...

Note:

n is positive and will fit within the range of a 32-bit signed integer ($n < 2^{31}$).

Example 1:

```
Input:
3

Output:
3
```

Example 2:

```
Input:
11

Output:
0

Explanation:
The 11th digit of the sequence 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11,
 ... is a 0, which is part of the number 10.
```

Companies

Google

Topics

Math

191. Number of 1 Bits (Easy)

Write a function that takes an unsigned integer and returns the number of '1' bits it has (also known as the Hamming weight (https://en.wikipedia.org/wiki/Hamming_weight)).

For example, the 32-bit integer '11' has binary representation `00000000000000000000000000001011`, so the function should return 3.

Companies

Microsoft   Apple

Topics

Bit Manipulation

323. Number of Connected Components in an Undirected Graph (Medium)

Given `n` nodes labeled from `0` to `n - 1` and a list of undirected edges (each edge is a pair of nodes), write a function to find the number of connected components in an undirected graph.

Example 1:

```
    0       3
    |       |
    1 --- 2  4
```

Given `n = 5` and `edges = [[0, 1], [1, 2], [3, 4]]`, return `2`.

Example 2:

```
    0           4
    |           |
    1 --- 2 --- 3
```

Given `n = 5` and `edges = [[0, 1], [1, 2], [2, 3], [3, 4]]`, return `1`.

Note:
You can assume that no duplicate edges will appear in `edges`. Since all edges are undirected, `[0, 1]` is the same as `[1, 0]` and thus will not appear together in `edges`.

Companies

Google   Twitter

Topics

Depth-first Search   Breadth-first Search   Union Find   Graph

233. Number of Digit One (Hard)

Given an integer n, count the total number of digit 1 appearing in all non-negative integers less than or equal to n.

For example:

Given n = 13,

Return 6, because digit 1 occurred in the following numbers: 1, 10, 11, 12, 13.

Companies

Topics

Math

305. Number of Islands II (Hard)

A 2d grid map of `m` rows and `n` columns is initially filled with water. We may perform an addLand operation which turns the water at position (row, col) into a land. Given a list of positions to operate, count the number of islands after each addLand operation . 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:

Given `m = 3`, `n = 3`, `positions = [[0,0], [0,1], [1,2], [2,1]]`.
Initially, the 2d grid `grid` is filled with water. (Assume 0 represents water and 1 represents land).

```
0 0 0
0 0 0
0 0 0
```

Operation #1: addLand(0, 0) turns the water at grid[0][0] into a land.

```
1 0 0
0 0 0    Number of islands = 1
0 0 0
```

Operation #2: addLand(0, 1) turns the water at grid[0][1] into a land.

```
1 1 0
0 0 0    Number of islands = 1
0 0 0
```

Operation #3: addLand(1, 2) turns the water at grid[1][2] into a land.

```
1 1 0
0 0 1     Number of islands = 2
0 0 0
```

Operation #4: addLand(2, 1) turns the water at grid[2][1] into a land.

```
1 1 0
0 0 1     Number of islands = 3
0 1 0
```

We return the result as an array: `[1, 1, 2, 3]`

Challenge:

Can you do it in time complexity O(k log mn), where k is the length of the `positions`?

Companies

(Google)

Topics

(Union Find)

200. Number of Islands (Medium)

Given a 2d grid map of `'1'`s (land) and `'0'`s (water), count 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 1:

```
11110
11010
11000
00000
```

Answer: 1

Example 2:

```
11000
11000
00100
00011
```

Answer: 3

Companies

Google   Facebook   Microsoft   Amazon   Zenefits

Topics

Depth-first Search   Breadth-first Search   Union Find

328. Odd Even Linked List (Medium)

Given a singly linked list, group all odd nodes together followed by the even nodes. Please note here we are talking about the node number and not the value in the nodes.

You should try to do it in place. The program should run in O(1) space complexity and O(nodes) time complexity.

Example:
Given `1->2->3->4->5->NULL`,
return `1->3->5->2->4->NULL`.

Note:
The relative order inside both the even and odd groups should remain as it was in the input.
The first node is considered odd, the second node even and so on ...

Companies

Topics

(Linked List)

161. One Edit Distance (Medium)

Given two strings S and T, determine if they are both one edit distance apart.

Companies

Facebook   Uber   Twitter   Snapchat

Topics

String

417. Pacific Atlantic Water Flow (Medium)

Given an `m x n` matrix of non-negative integers representing the height of each unit cell in a continent, the "Pacific ocean" touches the left and top edges of the matrix and the "Atlantic ocean" touches the right and bottom edges.

Water can only flow in four directions (up, down, left, or right) from a cell to another one with height equal or lower.

Find the list of grid coordinates where water can flow to both the Pacific and Atlantic ocean.

Note:

1. The order of returned grid coordinates does not matter.
2. Both m and n are less than 150.

Example:

```
Given the following 5x5 matrix:

  Pacific ~    ~    ~    ~    ~
       ~   1    2    2    3   (5)  *
       ~   3    2    3   (4)  (4)  *
       ~   2    4   (5)   3    1   *
       ~  (6)  (7)   1    4    5   *
       ~  (5)   1    1    2    4   *
           *    *    *    *    *  Atlantic

Return:

[[0, 4], [1, 3], [1, 4], [2, 2], [3, 0], [3, 1], [4, 0]] (positions
 with parentheses in above matrix).
```

Companies

Google

Topics

Depth-first Search   Breadth-first Search

276. Paint Fence (Easy)

There is a fence with n posts, each post can be painted with one of the k colors.

You have to paint all the posts such that no more than two adjacent fence posts have the same color.

Return the total number of ways you can paint the fence.

Note:
n and k are non-negative integers.

Companies

Google

Topics

Dynamic Programming

265. Paint House II (Hard)

There are a row of n houses, each house can be painted with one of the k colors. The cost of painting each house with a certain color is different. You have to paint all the houses such that no two adjacent houses have the same color.

The cost of painting each house with a certain color is represented by a `n x k` cost matrix. For example, `costs[0][0]` is the cost of painting house 0 with color 0; `costs[1][2]` is the cost of painting house 1 with color 2, and so on... Find the minimum cost to paint all houses.

Note:
All costs are positive integers.

Follow up:
Could you solve it in O(nk) runtime?

Companies

Facebook

Topics

Dynamic Programming

256. Paint House (Easy)

There are a row of n houses, each house can be painted with one of the three colors: red, blue or green. The cost of painting each house with a certain color is different. You have to paint all the houses such that no two adjacent houses have the same color.

The cost of painting each house with a certain color is represented by a `n x 3` cost matrix. For example, `costs[0][0]` is the cost of painting house 0 with color red; `costs[1][2]` is the cost of painting house 1 with color green, and so on... Find the minimum cost to paint all houses.

Note:
All costs are positive integers.

Companies

LinkedIn

Topics

Dynamic Programming

234. Palindrome Linked List (Easy)

Given a singly linked list, determine if it is a palindrome.

Follow up:

Could you do it in O(n) time and O(1) space?

Companies

Facebook   Amazon   IXL

Topics

Linked List   Two Pointers

9. Palindrome Number (Easy)

Determine whether an integer is a palindrome. Do this without extra space.

Some hints:

Could negative integers be palindromes? (ie, -1)

If you are thinking of converting the integer to string, note the restriction of using extra space.

You could also try reversing an integer. However, if you have solved the problem "Reverse Integer", you know that the reversed integer might overflow. How would you handle such case?

There is a more generic way of solving this problem.

Companies

Topics

Math

Loading [MathJax]/extensions/MathZoom.js

336. Palindrome Pairs (Hard)

Given a list of unique words, find all pairs of distinct indices `(i, j)` in the given list, so that the concatenation of the two words, i.e. `words[i] + words[j]` is a palindrome.

Example 1:
Given `words` = `["bat", "tab", "cat"]`
Return `[[0, 1], [1, 0]]`
The palindromes are `["battab", "tabbat"]`

Example 2:
Given `words` = `["abcd", "dcba", "lls", "s", "sssll"]`
Return `[[0, 1], [1, 0], [3, 2], [2, 4]]`
The palindromes are `["dcbaabcd", "abcddcba", "slls", "llssssll"]`

Companies

Google   Airbnb

Topics

Hash Table   String   Trie

132. Palindrome Partitioning II (Hard)

Given a string s, partition s such that every substring of the partition is a palindrome.

Return the minimum cuts needed for a palindrome partitioning of s.

For example, given s = `"aab"`,
Return `1` since the palindrome partitioning `["aa","b"]` could be produced using 1 cut.

Companies

Topics

Dynamic Programming

131. Palindrome Partitioning (Medium)

Given a string s, partition s such that every substring of the partition is a palindrome.

Return all possible palindrome partitioning of s.

For example, given s = `"aab"`,
Return

```
[
  ["aa","b"],
  ["a","a","b"]
]
```

Companies

Bloomberg

Topics

Backtracking

267. Palindrome Permutation II (Medium)

Given a string `s`, return all the palindromic permutations (without duplicates) of it. Return an empty list if no palindromic permutation could be form.

For example:

Given `s` = `"aabb"`, return `["abba", "baab"]`.

Given `s` = `"abc"`, return `[]`.

Companies

Topics

Backtracking

266. Palindrome Permutation (Easy)

Given a string, determine if a permutation of the string could form a palindrome.

For example,
`"code"` -> False, `"aab"` -> True, `"carerac"` -> True.

Companies

Google   Bloomberg   Uber

Topics

Hash Table

416. Partition Equal Subset Sum (Medium)

Given a non-empty array containing only positive integers , find if the array can be partitioned into two subsets such that the sum of elements in both subsets is equal.

Note:

1. Each of the array element will not exceed 100.
2. The array size will not exceed 200.

Example 1:

```
Input: [1, 5, 11, 5]

Output: true

Explanation: The array can be partitioned as [1, 5, 5] and [11].
```

Example 2:

```
Input: [1, 2, 3, 5]

Output: false

Explanation: The array cannot be partitioned into equal sum subsets.
```

Companies

eBay

Topics

Dynamic Programming

86. Partition List (Medium)

Given a linked list and a value x, partition it such that all nodes less than x come before nodes greater than or equal to x.

You should preserve the original relative order of the nodes in each of the two partitions.

For example,
Given `1->4->3->2->5->2` and x = 3,
return `1->2->2->4->3->5`.

Companies

Topics

Linked List   Two Pointers

119. Pascal's Triangle II (Easy)

Given an index k, return the kth row of the Pascal's triangle.

For example, given k = 3,
Return `[1,3,3,1]`.

Note:
Could you optimize your algorithm to use only O(k) extra space?

Companies

Amazon

Topics

Array

118. Pascal's Triangle (Easy)

Given numRows, generate the first numRows of Pascal's triangle.

For example, given numRows = 5,
Return

```
[
     [1],
    [1,1],
   [1,2,1],
  [1,3,3,1],
 [1,4,6,4,1]
]
```

Companies

Twitter   Apple

Topics

Array

330. Patching Array (Hard)

Given a sorted positive integer array nums and an integer n, add/patch elements to the array such that any number in range `[1, n]` inclusive can be formed by the sum of some elements in the array. Return the minimum number of patches required.

Example 1:
nums = `[1, 3]`, n = 6
Return `1`.

Combinations of nums are `[1]`, `[3]`, `[1,3]`, which form possible sums of: `1, 3, 4`.
Now if we add/patch `2` to nums, the combinations are: `[1]`, `[2]`, `[3]`, `[1,3]`, `[2,3]`, `[1,2,3]`.
Possible sums are `1, 2, 3, 4, 5, 6`, which now covers the range `[1, 6]`.
So we only need `1` patch.

Example 2:
nums = `[1, 5, 10]`, n = 20
Return `2`.
The two patches can be `[2, 4]`.

Example 3:
nums = `[1, 2, 2]`, n = 5
Return `0`.

Companies

Google

Topics

Greedy

113. Path Sum II (Medium)

Given a binary tree and a sum, find all root-to-leaf paths where each path's sum equals the given sum.

For example:
Given the below binary tree and `sum = 22`,

```
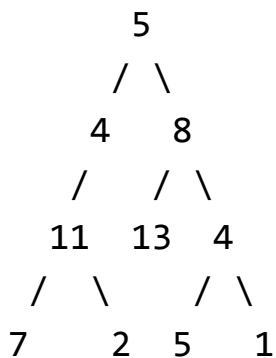              5
             / \
            4   8
           /   / \
         11  13   4
         / \     / \
        7   2   5   1
```

return

```
[
   [5,4,11,2],
   [5,8,4,5]
]
```

Companies

Bloomberg

Topics

Tree   Depth-first Search

112. Path Sum (Easy)

Given a binary tree and a sum, determine if the tree has a root-to-leaf path such that adding up all the values along the path equals the given sum.

For example:
Given the below binary tree and `sum = 22`,

```
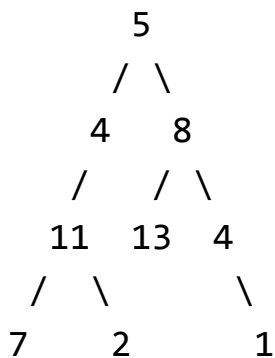          5
         / \
        4   8
       /   / \
      11  13  4
     /  \      \
    7    2      1
```

return true, as there exist a root-to-leaf path `5->4->11->2` which sum is 22.

Companies

Microsoft

Topics

Tree   Depth-first Search

Loading [MathJax]/extensions/AssistiveMML.js

284. Peeking Iterator (Medium)

Given an Iterator class interface with methods: `next()` and `hasNext()`, design and implement a PeekingIterator that support the `peek()` operation -- it essentially peek() at the element that will be returned by the next call to next().

Here is an example. Assume that the iterator is initialized to the beginning of the list: `[1, 2, 3]`.

Call `next()` gets you 1, the first element in the list.

Now you call `peek()` and it returns 2, the next element. Calling `next()` after that still return 2.

You call `next()` the final time and it returns 3, the last element. Calling `hasNext()` after that should return false.

Follow up : How would you extend your design to be generic and work with all types, not just integer?

Companies

Google   Apple   Yahoo

Topics

Design

391. Perfect Rectangle (Hard)

Given N axis-aligned rectangles where N > 0, determine if they all together form an exact cover of a rectangular region.

Each rectangle is represented as a bottom-left point and a top-right point. For example, a unit square is represented as [1,1,2,2]. (coordinate of bottom-left point is (1, 1) and top-right point is (2, 2)).

Example 1:


```
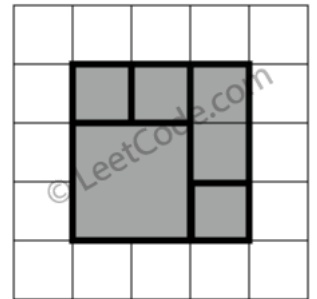rectangles = [
  [1,1,3,3],
  [3,1,4,2],
  [3,2,4,4],
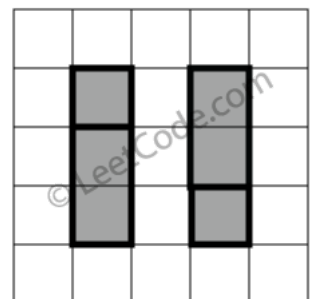  [1,3,2,4],
  [2,3,3,4]
]
```

Return true. All 5 rectangles together form an exact cover of a rectangular region.

Example 2:


```
rectangles = [
  [1,1,2,3],
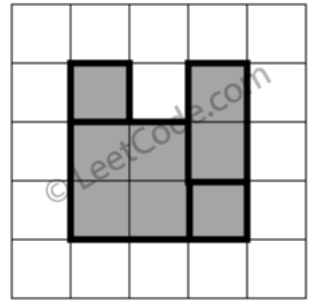  [1,3,2,4],
  [3,1,4,2],
  [3,2,4,4]
]
```

Return false. Because there is a gap between the two rectangular regions.

Example 3:

```
rectangles = [
  [1,1,3,3],
  [3,1,4,2],
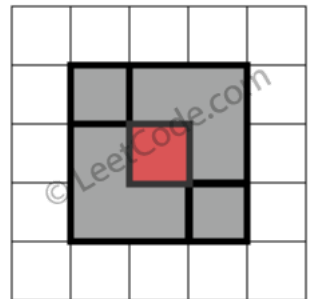  [1,3,2,4],
  [3,2,4,4]
]


Return false. Because there is a gap in the top cent
er.
```


Example 4:

```
rectangles = [
  [1,1,3,3],
  [3,1,4,2],
  [1,3,2,4],
  [2,2,4,4]
]


Return false. Because two of the rectangles overlap
 with each other.
```


Companies

Google

Topics

279. Perfect Squares (Medium)

Given a positive integer n, find the least number of perfect square numbers (for example, `1, 4, 9, 16, ...`) which sum to n.

For example, given n = `12`, return `3` because `12 = 4 + 4 + 4`; given n = `13`, return `2` because `13 = 4 + 9`.

Companies

Google

Topics

Math   Dynamic Programming   Breadth-first Search

60. Permutation Sequence (Medium)

The set `[1,2,3,…,n]` contains a total of n! unique permutations.

By listing and labeling all of the permutations in order,
We get the following sequence (ie, for n = 3):

1. `"123"`
2. `"132"`
3. `"213"`
4. `"231"`
5. `"312"`
6. `"321"`

Given n and k, return the kth permutation sequence.

Note: Given n will be between 1 and 9 inclusive.

Companies

Twitter

Topics

Math   Backtracking

47. Permutations II (Medium)

Given a collection of numbers that might contain duplicates, return all possible unique permutations.

For example,
`[1,1,2]` have the following unique permutations:

```
[
  [1,1,2],
  [1,2,1],
  [2,1,1]
]
```

Companies

Microsoft   LinkedIn

Topics

Backtracking

46. Permutations (Medium)

Given a collection of distinct numbers, return all possible permutations.

For example,
`[1,2,3]` have the following permutations:

```
[
  [1,2,3],
  [1,3,2],
  [2,1,3],
  [2,3,1],
  [3,1,2],
  [3,2,1]
]
```

Companies

Microsoft   LinkedIn

Topics

Backtracking

Loading [MathJax]/extensions/MathZoom.js

369. Plus One Linked List (Medium)

Given a non-negative integer represented as non-empty a singly linked list of digits, plus one to the integer.

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

The digits are stored such that the most significant digit is at the head of the list.

Example:

```
Input:
1->2->3

Output:
1->2->4
```

Companies

Google

Topics

Linked List

66. Plus One (Easy)

Given a non-negative integer represented as a non-empty array of digits, plus one to the integer.

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

The digits are stored such that the most significant digit is at the head of the list.

Companies

Google

Topics

Array   Math

117. Populating Next Right Pointers in Each Node II (Medium)

Follow up for problem "Populating Next Right Pointers in Each Node".

What if the given tree could be any binary tree? Would your previous solution still work?

Note:

- You may only use constant extra space.

For example,
Given the following binary tree,

```
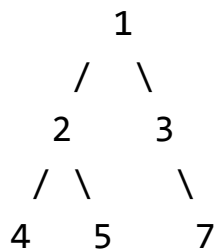        1
       /  \
      2    3
     / \    \
    4   5    7
```

After calling your function, the tree should look like:

```
          1 -> NULL
         /  \
        2 -> 3 -> NULL
       / \    \
      4-> 5 -> 7 -> NULL
```

Companies

Facebook   Microsoft   Bloomberg

Topics

Tree   Depth-first Search

116. Populating Next Right Pointers in Each Node (Medium)

Given a binary tree

```
struct TreeLinkNode {
  TreeLinkNode *left;
  TreeLinkNode *right;
  TreeLinkNode *next;
}
```

Populate each next pointer to point to its next right node. If there is no next right node, the next pointer should be set to `NULL`.

Initially, all next pointers are set to `NULL`.

Note:

- You may only use constant extra space.
- You may assume that it is a perfect binary tree (ie, all leaves are at the same level, and every parent has two children).

For example,
Given the following perfect binary tree,

```
       1
     /   \
    2      3
   / \    / \
  4   5  6   7
```

After calling your function, the tree should look like:

Loading [MathJax]/extensions/MathZoom.js

```
     1 -> NULL
   /  \
  2 -> 3 -> NULL
 / \  / \
4->5->6->7 -> NULL
```

Companies

Microsoft

Topics

Tree   Depth-first Search

50. Pow(x, n) (Medium)

Implement pow(x, n) (http://www.cplusplus.com/reference/valarray/pow/).

Example 1:

```
Input: 2.00000, 10
Output: 1024.00000
```

Example 2:

```
Input: 2.10000, 3
Output: 9.26100
```

Companies

Google   Facebook   Bloomberg   LinkedIn

Topics

Math   Binary Search

342. Power of Four (Easy)

Given an integer (signed 32 bits), write a function to check whether it is a power of 4.

Example:
Given num = 16, return true. Given num = 5, return false.

Follow up : Could you solve it without loops/recursion?

Companies

(Two Sigma)

Topics

(Bit Manipulation)

326. Power of Three (Easy)

Given an integer, write a function to determine if it is a power of three.

Follow up:
Could you do it without using any loop / recursion?

Companies

Google

Topics

Math

231. Power of Two (Easy)

Given an integer, write a function to determine if it is a power of two.

Companies

Google

Topics

Math   Bit Manipulation

238. Product of Array Except Self (Medium)

Given an array of n integers where n > 1, `nums`, return an array `output` such that `output[i]` is equal to the product of all the elements of `nums` except `nums[i]`.

Solve it without division   and in O(n).

For example, given `[1,2,3,4]`, return `[24,12,8,6]`.

Follow up:
Could you solve it with constant space complexity? (Note: The output array does not count as extra space for the purpose of space complexity analysis.)

Companies

Facebook   Microsoft   Amazon   LinkedIn   Apple

Topics

Array

406. Queue Reconstruction by Height (Medium)

Suppose you have a random list of people standing in a queue. Each person is described by a pair of integers (h, k), where h is the height of the person and k is the number of people in front of this person who have a height greater than or equal to h. Write an algorithm to reconstruct the queue.

Note:
The number of people is less than 1,100.

Example

```
Input:
[[7,0], [4,4], [7,1], [5,0], [6,1], [5,2]]

Output:
[[5,0], [7,0], [5,2], [6,1], [4,4], [7,1]]
```

Companies

Google

Topics

Greedy

398. Random Pick Index (Medium)

Given an array of integers with possible duplicates, randomly output the index of a given target number. You can assume that the given target number must exist in the array.

Note:
The array size can be very large. Solution that uses too much extra space will not pass the judge.

Example:

```
int[] nums = new int[] {1,2,3,3,3};
Solution solution = new Solution(nums);

// pick(3) should return either index 2, 3, or 4 randomly. Each index should have equal probability of returning.
solution.pick(3);

// pick(1) should return 0. Since in the array only nums[0] is equal to 1.
solution.pick(1);
```

Companies

Facebook

Topics

Reservoir Sampling

370. Range Addition (Medium)

Assume you have an array of length n initialized with all 0's and are given k update operations.

Each operation is represented as a triplet: [startIndex, endIndex, inc]   which increments each element of subarray A[startIndex ... endIndex]   (startIndex and endIndex inclusive) with inc .

Return the modified array after all k operations were executed.

Example:

```
Given:

    length = 5,
    updates = [
        [1,  3,  2],
        [2,  4,  3],
        [0,  2, -2]
    ]

Output:

    [-2, 0, 3, 5, 3]
```

Explanation:

```
Initial state:
[ 0, 0, 0, 0, 0 ]

After applying operation [1, 3, 2]:
[ 0, 2, 2, 2, 0 ]

After applying operation [2, 4, 3]:
[ 0, 2, 5, 5, 3 ]

After applying operation [0, 2, -2]:
[-2, 0, 3, 5, 3 ]
```

Companies

Google

Topics

Array

303. Range Sum Query - Immutable (Easy)

Given an integer array nums, find the sum of the elements between indices i and j (i ≤ j), inclusive.

Example:

```
Given nums = [-2, 0, 3, -5, 2, -1]

sumRange(0, 2) -> 1
sumRange(2, 5) -> -1
sumRange(0, 5) -> -3
```

Note:

1. You may assume that the array does not change.
2. There are many calls to sumRange function.

Companies

Palantir

Topics

Dynamic Programming

307. Range Sum Query - Mutable (Medium)

Given an integer array nums, find the sum of the elements between indices i and j (i ≤ j), inclusive.

The update(i, val) function modifies nums by updating the element at index i to val.

Example:

```
Given nums = [1, 3, 5]

sumRange(0, 2) -> 9
update(1, 2)
sumRange(0, 2) -> 8
```

Note:

1. The array is only modifiable by the update function.
2. You may assume the number of calls to update and sumRange function is distributed evenly.

Companies

Topics

Binary Indexed Tree   Segment Tree

304. Range Sum Query 2D - Immutable (Medium)

Given a 2D matrix matrix, find the sum of the elements inside the rectangle defined by its upper left corner (row1, col1) and lower right corner (row2, col2).

3	0	1	4	2
5	6	3	2	1
1	2	0	1	5
4	1	0	1	7
1	0	3	0	5

The above rectangle (with the red border) is defined by (row1, col1) = (2, 1) and (row2, col2) = (4, 3), which contains sum = 8.

Example:

```
Given matrix = [
  [3, 0, 1, 4, 2],
  [5, 6, 3, 2, 1],
  [1, 2, 0, 1, 5],
  [4, 1, 0, 1, 7],
  [1, 0, 3, 0, 5]
]

sumRegion(2, 1, 4, 3) -> 8
sumRegion(1, 1, 2, 2) -> 11
sumRegion(1, 2, 2, 4) -> 12
```

Note:

1. You may assume that the matrix does not change.
2. There are many calls to sumRegion function.
3. You may assume that row1 ≤ row2 and col1 ≤ col2.

Companies

Topics

(Dynamic Programming)

308. Range Sum Query 2D - Mutable (Hard)

Given a 2D matrix matrix, find the sum of the elements inside the rectangle defined by its upper left corner (row1, col1) and lower right corner (row2, col2).

3	0	1	4	2
5	6	3	2	1
1	2	0	1	5
4	1	0	1	7
1	0	3	0	5

The above rectangle (with the red border) is defined by (row1, col1) = (2, 1) and (row2, col2) = (4, 3), which contains sum = 8.

Example:

```
Given matrix = [
  [3, 0, 1, 4, 2],
  [5, 6, 3, 2, 1],
  [1, 2, 0, 1, 5],
  [4, 1, 0, 1, 7],
  [1, 0, 3, 0, 5]
]

sumRegion(2, 1, 4, 3) -> 8
update(3, 2, 2)
sumRegion(2, 1, 4, 3) -> 10
```

Note:

1. The matrix is only modifiable by the update function.
2. You may assume the number of calls to update and sumRegion function is distributed evenly.
3. You may assume that row1 ≤ row2 and col1 ≤ col2.

Companies

Google

Topics

Binary Indexed Tree Segment Tree

383. Ransom Note (Easy)

Given an arbitrary ransom note string and another string containing letters from all the magazines, write a function that will return true if the ransom note can be constructed from the magazines ; otherwise, it will return false.

Each letter in the magazine string can only be used once in your ransom note.

Note:
You may assume that both strings contain only lowercase letters.

```
canConstruct("a", "b") -> false
canConstruct("aa", "ab") -> false
canConstruct("aa", "aab") -> true
```

Companies

Apple

Topics

String

158. Read N Characters Given Read4 II - Call multiple times (Hard)

The API: `int read4(char *buf)` reads 4 characters at a time from a file.

The return value is the actual number of characters read. For example, it returns 3 if there is only 3 characters left in the file.

By using the `read4` API, implement the function `int read(char *buf, int n)` that reads n characters from the file.

Note:
The `read` function may be called multiple times.

Companies

Google   Facebook   Bloomberg

Topics

String

157. Read N Characters Given Read4 (Easy)

The API: `int read4(char *buf)` reads 4 characters at a time from a file.

The return value is the actual number of characters read. For example, it returns 3 if there is only 3 characters left in the file.

By using the `read4` API, implement the function `int read(char *buf, int n)` that reads n characters from the file.

Note:
The `read` function will only be called once for each test case.

Companies

(Facebook)

Topics

(String)

358. Rearrange String k Distance Apart (Hard)

Given a non-empty string s and an integer k, rearrange the string such that the same characters are at least distance k from each other.

All input strings are given in lowercase letters. If it is not possible to rearrange the string, return an empty string `""`.

Example 1:

```
s = "aabbcc", k = 3

Result: "abcabc"

The same letters are at least distance 3 from each other.
```

Example 2:

```
s = "aaabc", k = 3

Answer: ""

It is not possible to rearrange the string.
```

Example 3:

```
s = "aaadbbcc", k = 2

Answer: "abacabcd"

Another possible answer is: "abcabcda"

The same letters are at least distance 2 from each other.
```

Companies

Google

Topics

Hash Table · Heap · Greedy

332. Reconstruct Itinerary (Medium)

Given a list of airline tickets represented by pairs of departure and arrival airports `[from, to]`, reconstruct the itinerary in order. All of the tickets belong to a man who departs from `JFK`. Thus, the itinerary must begin with `JFK`.

Note:

1. If there are multiple valid itineraries, you should return the itinerary that has the smallest lexical order when read as a single string. For example, the itinerary `["JFK", "LGA"]` has a smaller lexical order than `["JFK", "LGB"]`.
2. All airports are represented by three capital letters (IATA code).
3. You may assume all tickets form at least one valid itinerary.

Example 1:
`tickets = [["MUC", "LHR"], ["JFK", "MUC"], ["SFO", "SJC"], ["LHR", "SFO"]]`
Return `["JFK", "MUC", "LHR", "SFO", "SJC"]`.

Example 2:
`tickets = [["JFK","SFO"],["JFK","ATL"],["SFO","ATL"],["ATL","JFK"], ["ATL","SFO"]]`
Return `["JFK","ATL","JFK","SFO","ATL","SFO"]`.
Another possible reconstruction is `["JFK","SFO","ATL","JFK","ATL","SFO"]`. But it is larger in lexical order.

Companies

(Google)

Topics

(Depth-first Search) (Graph)

99. Recover Binary Search Tree (Hard)

Two elements of a binary search tree (BST) are swapped by mistake.

Recover the tree without changing its structure.

Note:
A solution using O(n) space is pretty straight forward. Could you devise a constant space solution?

Companies

Topics

(Tree) (Depth-first Search)

223. Rectangle Area (Medium)

Find the total area covered by two rectilinear rectangles in a 2D plane.

Each rectangle is defined by its bottom left corner and top right corner as shown in the figure.

Assume that the total area is never beyond the maximum possible value of int .

Companies

Topics

Math

10. Regular Expression Matching (Hard)

Implement regular expression matching with support for '.' and '*'.

```
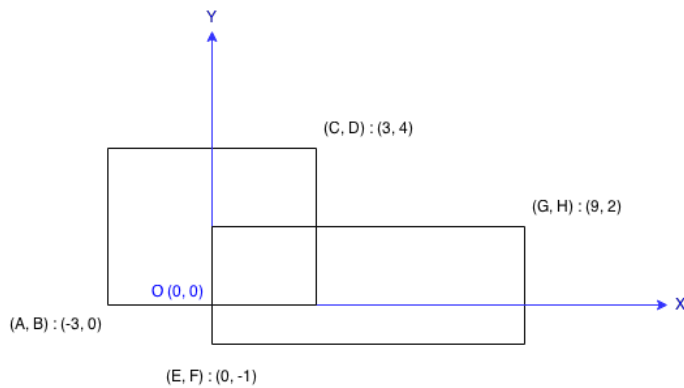'.' Matches any single character.
'*' Matches zero or more of the preceding element.

The matching should cover the entire input string (not partial).

The function prototype should be:
bool isMatch(const char *s, const char *p)

Some examples:
isMatch("aa","a") → false
isMatch("aa","aa") → true
isMatch("aaa","aa") → false
isMatch("aa", "a*") → true
isMatch("aa", ".*") → true
isMatch("ab", ".*") → true
isMatch("aab", "c*a*b") → true
```

Companies

Google   Facebook   Uber   Twitter   Airbnb

Topics

String   Dynamic Programming   Backtracking

316. Remove Duplicate Letters (Hard)

Given a string which contains only lowercase letters, remove duplicate letters so that every letter appear once and only once. You must make sure your result is the smallest in lexicographical order among all possible results.

Example:

Given `"bcabc"`
Return `"abc"`

Given `"cbacdcbc"`
Return `"acdb"`

Companies

(Google)

Topics

(Stack) (Greedy)

80. Remove Duplicates from Sorted Array II (Medium)

Follow up for "Remove Duplicates":
What if duplicates are allowed at most twice?

For example,
Given sorted array nums = `[1,1,1,2,2,3]`,

Your function should return length = `5`, with the first five elements of nums being `1`, `1`, `2`, `2` and `3`. It doesn't matter what you leave beyond the new length.

Companies

Facebook

Topics

Array   Two Pointers

26. Remove Duplicates from Sorted Array (Easy)

Given a sorted array, remove the duplicates in-place (https://en.wikipedia.org/wiki/In-place_algorithm) such that each element appear only once and return the new length.

Do not allocate extra space for another array, you must do this by modifying the input array in-place (https://en.wikipedia.org/wiki/In-place_algorithm)   with O(1) extra memory.

Example:

```
Given nums = [1,1,2],

Your function should return length = 2, with the first two elements
 of nums being 1 and 2 respectively.
It doesn't matter what you leave beyond the new length.
```

Companies

Facebook   Microsoft   Bloomberg

Topics

Array   Two Pointers

82. Remove Duplicates from Sorted List II (Medium)

Given a sorted linked list, delete all nodes that have duplicate numbers, leaving only distinct numbers from the original list.

For example,
Given `1->2->3->3->4->4->5`, return `1->2->5`.
Given `1->1->1->2->3`, return `2->3`.

Companies

Topics

Linked List

83. Remove Duplicates from Sorted List (Easy)

Given a sorted linked list, delete all duplicates such that each element appear only once.

For example,
Given `1->1->2`, return `1->2`.
Given `1->1->2->3->3`, return `1->2->3`.

Companies

Topics

Linked List

27. Remove Element (Easy)

Given an array and a value, remove all instances of that value in-place (https://en.wikipedia.org/wiki/In-place_algorithm) and return the new length.

Do not allocate extra space for another array, you must do this by modifying the input array in-place (https://en.wikipedia.org/wiki/In-place_algorithm) with O(1) extra memory.

The order of elements can be changed. It doesn't matter what you leave beyond the new length.

Example:

```
Given nums = [3,2,2,3], val = 3,

Your function should return length = 2, with the first two elements
 of nums being 2.
```

Companies

Topics

Array   Two Pointers

301. Remove Invalid Parentheses (Hard)

Remove the minimum number of invalid parentheses in order to make the input string valid. Return all possible results.

Note: The input string may contain letters other than the parentheses (and).

Examples:

```
"()())()" -> ["()()()", "(())()"]
"(a)())()" -> ["(a)()()", "(a())()"]
")(" -> [""]
```

Companies

Facebook

Topics

Depth-first Search   Breadth-first Search

402. Remove K Digits (Medium)

Given a non-negative integer num represented as a string, remove k digits from the number so that the new number is the smallest possible.

Note:

- The length of num is less than 10002 and will be ≥ k.
- The given num does not contain any leading zero.

Example 1:

```
Input: num = "1432219", k = 3
Output: "1219"
Explanation: Remove the three digits 4, 3, and 2 to form the new num
ber 1219 which is the smallest.
```

Example 2:

```
Input: num = "10200", k = 1
Output: "200"
Explanation: Remove the leading 1 and the number is 200. Note that t
he output must not contain leading zeroes.
```

Example 3:

```
Input: num = "10", k = 2
Output: "0"
Explanation: Remove all the digits from the number and it is left wi
th nothing which is 0.
```

Companies

Google   Snapchat

Topics

Stack   Greedy

203. Remove Linked List Elements (Easy)

Remove all elements from a linked list of integers that have value val.

Example

Given: 1 --> 2 --> 6 --> 3 --> 4 --> 5 --> 6, val = 6

Return: 1 --> 2 --> 3 --> 4 --> 5

Companies

Topics

Linked List

19. Remove Nth Node From End of List (Medium)

Given a linked list, remove the nth node from the end of list and return its head.

For example,

```
    Given linked list: 1->2->3->4->5, and n = 2.

    After removing the second node from the end, the linked list beco
mes 1->2->3->5.
```

Note:
Given n will always be valid.
Try to do this in one pass.

Companies

Topics

Linked List   Two Pointers

143. Reorder List (Medium)

Given a singly linked list L: $L_0 \rightarrow L_1 \rightarrow \ldots \rightarrow L_{n-1} \rightarrow L_n$,

reorder it to: $L_0 \rightarrow L_n \rightarrow L_1 \rightarrow L_{n-1} \rightarrow L_2 \rightarrow L_{n-2} \rightarrow \ldots$

You must do this in-place without altering the nodes' values.

For example,

Given `{1,2,3,4}`, reorder it to `{1,4,2,3}`.

Companies

Topics

Linked List

187. Repeated DNA Sequences (Medium)

All DNA is composed of a series of nucleotides abbreviated as A, C, G, and T, for example: "ACGAATTCCG". When studying DNA, it is sometimes useful to identify repeated sequences within the DNA.

Write a function to find all the 10-letter-long sequences (substrings) that occur more than once in a DNA molecule.

For example,

```
Given s = "AAAAACCCCCAAAAACCCCCCAAAAAGGGTTT",

Return:
["AAAAACCCCC", "CCCCCAAAAA"].
```

Companies

LinkedIn

Topics

Hash Table   Bit Manipulation

93. Restore IP Addresses (Medium)

Given a string containing only digits, restore it by returning all possible valid IP address combinations.

For example:
Given `"25525511135"`,

return `["255.255.11.135", "255.255.111.35"]`. (Order does not matter)

Companies

Topics

String   Backtracking

190. Reverse Bits (Easy)

Reverse bits of a given 32 bits unsigned integer.

For example, given input 43261596 (represented in binary as 00000010100101000001111010011100), return 964176192 (represented in binary as 00111001011110000010100101000000).

Follow up：
If this function is called many times, how would you optimize it?

Related problem: Reverse Integer (/problems/reverse-integer/)

Companies

(Apple) (Airbnb)

Topics

(Bit Manipulation)

7. Reverse Integer (Easy)

Given a 32-bit signed integer, reverse digits of an integer.

Example 1:

```
Input: 123
Output:  321
```

Example 2:

```
Input: -123
Output: -321
```

Example 3:

```
Input: 120
Output: 21
```

Note:
Assume we are dealing with an environment which could only hold integers within the 32-bit signed integer range. For the purpose of this problem, assume that your function returns 0 when the reversed integer overflows.

Companies

Bloomberg   Apple

Topics

Math

92. Reverse Linked List II (Medium)

Reverse a linked list from position m to n. Do it in-place and in one-pass.

For example:

Given `1->2->3->4->5->NULL`, m = 2 and n = 4,

return `1->4->3->2->5->NULL`.

Note:

Given m, n satisfy the following condition:

1 ≤ m ≤ n ≤ length of list.

Companies

Topics

(Linked List)

206. Reverse Linked List (Easy)

Reverse a singly linked list.

Hint:

A linked list can be reversed either iteratively or recursively. Could you implement both?

Companies

Facebook Microsoft Amazon Bloomberg Uber Twitter Apple Snapchat Zenefits Yelp

Yahoo Adobe

Topics

Linked List

25. Reverse Nodes in k-Group (Hard)

Given a linked list, reverse the nodes of a linked list k at a time and return its modified list.

k is a positive integer and is less than or equal to the length of the linked list. If the number of nodes is not a multiple of k then left-out nodes in the end should remain as it is.

You may not alter the values in the nodes, only nodes itself may be changed.

Only constant memory is allowed.

For example,
Given this linked list: `1->2->3->4->5`

For k = 2, you should return: `2->1->4->3->5`

For k = 3, you should return: `3->2->1->4->5`

Companies

Facebook   Microsoft

Topics

Linked List

344. Reverse String (Easy)

Write a function that takes a string as input and returns the string reversed.

Example:

Given s = "hello", return "olleh".

Companies

Topics

Two Pointers   String

345. Reverse Vowels of a String (Easy)

Write a function that takes a string as input and reverse only the vowels of a string.

Example 1:
Given s = "hello", return "holle".

Example 2:
Given s = "leetcode", return "leotcede".

Note:
The vowels does not include the letter "y".

Companies

Google

Topics

Two Pointers   String

186. Reverse Words in a String II (Medium)

Given an input string, reverse the string word by word. A word is defined as a sequence of non-space characters.

The input string does not contain leading or trailing spaces and the words are always separated by a single space.

For example,
Given s = "`the sky is blue`",
return "`blue is sky the`".

Could you do it in-place without allocating extra space?

Related problem: Rotate Array (/problems/rotate-array/)

Update (2017-10-16):
We have updated the function signature to accept a `character array`, so please reset to the default code definition by clicking on the reload button above the code editor. Also, Run Code is now available!

Companies

(Microsoft) (Amazon) (Uber)

Topics

(String)

151. Reverse Words in a String (Medium)

Given an input string, reverse the string word by word.

For example,
Given s = "`the sky is blue`",
return "`blue is sky the`".

Update (2015-02-12):
For C programmers: Try to solve it in-place in O(1) space.

Clarification:

- What constitutes a word?

  A sequence of non-space characters constitutes a word.

- Could the input string contain leading or trailing spaces?

  Yes. However, your reversed string should not contain leading or trailing spaces.

- How about multiple spaces between two words?

  Reduce them to a single space in the reversed string.

Companies

Microsoft   Bloomberg   Apple   Snapchat   Yelp

Topics

String

13. Roman to Integer (Easy)

Given a roman numeral, convert it to an integer.

Input is guaranteed to be within the range from 1 to 3999.

Companies

Facebook   Microsoft   Bloomberg   Uber   Yahoo

Topics

Math   String

189. Rotate Array (Easy)

Rotate an array of n elements to the right by k steps.

For example, with n = 7 and k = 3, the array `[1,2,3,4,5,6,7]` is rotated to `[5,6,7,1,2,3,4]`.

Note:
Try to come up as many solutions as you can, there are at least 3 different ways to solve this problem.

Hint:

Could you do it in-place with O(1) extra space?

Related problem: Reverse Words in a String II (/problems/reverse-words-in-a-string-ii/)

Companies

Microsoft   Amazon   Bloomberg

Topics

Array

396. Rotate Function (Medium)

Given an array of integers `A` and let n to be its length.

Assume B_k to be an array obtained by rotating the array `A` k positions clock-wise, we define a "rotation function" `F` on `A` as follow:

`F(k)` = 0 * B_k`[0]` + 1 * B_k`[1]` + ... + (n-1) * B_k`[n-1]`.

Calculate the maximum value of `F(0)`, `F(1)`, ..., `F(n-1)`.

Note:
n is guaranteed to be less than 10^5.

Example:

```
A = [4, 3, 2, 6]

F(0) = (0 * 4) + (1 * 3) + (2 * 2) + (3 * 6) = 0 + 3 + 4 + 18 = 25
F(1) = (0 * 6) + (1 * 4) + (2 * 3) + (3 * 2) = 0 + 4 + 6 + 6 = 16
F(2) = (0 * 2) + (1 * 6) + (2 * 4) + (3 * 3) = 0 + 6 + 8 + 9 = 23
F(3) = (0 * 3) + (1 * 2) + (2 * 6) + (3 * 4) = 0 + 2 + 12 + 12 = 26

So the maximum value of F(0), F(1), F(2), F(3) is F(3) = 26.
```

Companies

Amazon

Topics

Math

48. Rotate Image (Medium)

You are given an n x n 2D matrix representing an image.

Rotate the image by 90 degrees (clockwise).

Note:
You have to rotate the image in-place , which means you have to modify the input 2D matrix directly. DO NOT allocate another 2D matrix and do the rotation.

Example 1:

```
Given input matrix =
[
  [1,2,3],
  [4,5,6],
  [7,8,9]
],

rotate the input matrix in-place such that it becomes:
[
  [7,4,1],
  [8,5,2],
  [9,6,3]
]
```

Example 2:

```
Given input matrix =
[
  [ 5, 1, 9,11],
  [ 2, 4, 8,10],
  [13, 3, 6, 7],
  [15,14,12,16]
],

rotate the input matrix in-place such that it becomes:
[
  [15,13, 2, 5],
  [14, 3, 4, 1],
  [12, 6, 8, 9],
  [16, 7,10,11]
]
```

Companies

Microsoft   Amazon   Apple

Topics

Array

61. Rotate List (Medium)

Given a list, rotate the list to the right by k places, where k is non-negative.

Example:

```
Given 1->2->3->4->5->NULL and k = 2,

return 4->5->1->2->3->NULL.
```

Companies

Topics

Linked List   Two Pointers

354. Russian Doll Envelopes (Hard)

You have a number of envelopes with widths and heights given as a pair of integers `(w, h)`. One envelope can fit into another if and only if both the width and height of one envelope is greater than the width and height of the other envelope.

What is the maximum number of envelopes can you Russian doll? (put one inside other)

Example:
Given envelopes = `[[5,4],[6,4],[6,7],[2,3]]`, the maximum number of envelopes you can Russian doll is `3` ([2,3] => [5,4] => [6,7]).

Companies

(Google)

Topics

(Binary Search) (Dynamic Programming)

100. Same Tree (Easy)

Given two binary trees, write a function to check if they are the same or not.

Two binary trees are considered the same if they are structurally identical and the nodes have the same value.

Example 1:

```
Input:      1          1
           / \        / \
          2   3      2   3
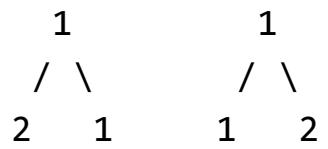
        [1,2,3],    [1,2,3]

Output: true
```

Example 2:

```
Input:      1          1
           /            \
          2              2

        [1,2],      [1,null,2]

Output: false
```

Example 3:

```
Input:     1          1
          / \        / \
         2   1      1   2

        [1,2,1],   [1,1,2]

Output: false
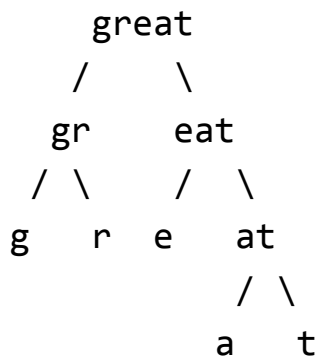```

## Companies

Bloomberg

## Topics

Tree   Depth-first Search

# 87. Scramble String (Hard)

Given a string s1, we may represent it as a binary tree by partitioning it to two non-empty substrings recursively.

Below is one possible representation of s1 = `"great"`:

```
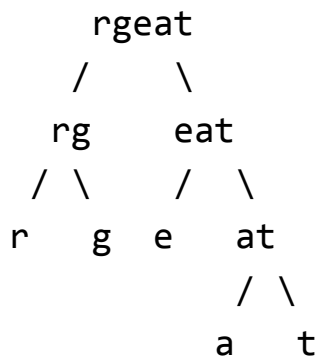    great
   /     \
  gr      eat
 / \     / \
g   r   e   at
           / \
          a   t
```

To scramble the string, we may choose any non-leaf node and swap its two children.

For example, if we choose the node `"gr"` and swap its two children, it produces a scrambled string `"rgeat"`.

```
    rgeat
   /     \
  rg      eat
 / \     / \
r   g   e   at
           / \
          a   t
```

We say that `"rgeat"` is a scrambled string of `"great"`.

Similarly, if we continue to swap the children of nodes `"eat"` and `"at"`, it produces a scrambled string `"rgtae"`.

```
    rgtae
   /    \
  rg     tae
 / \    / \
r   g  ta  e
      / \
     t   a
```

We say that `"rgtae"` is a scrambled string of `"great"`.

Given two strings s1 and s2 of the same length, determine if s2 is a scrambled string of s1.

Companies

Topics

(String) (Dynamic Programming)

240. Search a 2D Matrix II (Medium)

Write an efficient algorithm that searches for a value in an m x n matrix. This matrix has the following properties:

- Integers in each row are sorted in ascending from left to right.
- Integers in each column are sorted in ascending from top to bottom.

For example,

Consider the following matrix:

```
[
  [1,    4,   7, 11, 15],
  [2,    5,   8, 12, 19],
  [3,    6,   9, 16, 22],
  [10, 13, 14, 17, 24],
  [18, 21, 23, 26, 30]
]
```

Given target = 5, return true.

Given target = 20, return false.

Companies

Google Amazon Apple

Topics

Binary Search Divide and Conquer

74. Search a 2D Matrix (Medium)

Write an efficient algorithm that searches for a value in an m x n matrix. This matrix has the following properties:

- Integers in each row are sorted from left to right.
- The first integer of each row is greater than the last integer of the previous row.

For example,

Consider the following matrix:

```
[
  [1,   3,  5,  7],
  [10, 11, 16, 20],
  [23, 30, 34, 50]
]
```

Given target = 3, return `true`.

Companies

Topics

Array   Binary Search

34. Search for a Range (Medium)

Given an array of integers sorted in ascending order, find the starting and ending position of a given target value.

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

If the target is not found in the array, return `[-1, -1]`.

For example,
Given `[5, 7, 7, 8, 8, 10]` and target value 8,
return `[3, 4]`.

Companies

LinkedIn

Topics

Array   Binary Search

81. Search in Rotated Sorted Array II (Medium)

Follow up for "Search in Rotated Sorted Array":

What if duplicates are allowed?

Would this affect the run-time complexity? How and why?

Suppose an array sorted in ascending order is rotated at some pivot unknown to you beforehand.

(i.e., `0 1 2 4 5 6 7` might become `4 5 6 7 0 1 2`).

Write a function to determine if a given target is in the array.

The array may contain duplicates.

Companies

Topics

Array   Binary Search

33. Search in Rotated Sorted Array (Medium)

Suppose an array sorted in ascending order is rotated at some pivot unknown to you beforehand.

(i.e., `0 1 2 4 5 6 7` might become `4 5 6 7 0 1 2`).

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.

Companies

Facebook   Microsoft   Bloomberg   Uber   LinkedIn

Topics

Array   Binary Search

35. Search Insert Position (Easy)

Given a sorted array and a target value, return the index if the target is found. If not, return the index where it would be if it were inserted in order.

You may assume no duplicates in the array.

Example 1:

```
Input: [1,3,5,6], 5
Output: 2
```

Example 2:

```
Input: [1,3,5,6], 2
Output: 1
```

Example 3:

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

Example 1:

```
Input: [1,3,5,6], 0
Output: 0
```

Companies

Topics

Array   Binary Search

335. Self Crossing (Hard)

You are given an array x of `n` positive numbers. You start at point `(0,0)` and moves `x[0]` metres to the north, then `x[1]` metres to the west, `x[2]` metres to the south, `x[3]` metres to the east and so on. In other words, after each move your direction changes counter-clockwise.

Write a one-pass algorithm with `O(1)` extra space to determine, if your path crosses itself, or not.

Example 1:

```
Given x =

 [2, 1, 1, 2]

,
?????
?   ?
???????>
    ?

Return true (self crossing)
```

Example 2:

```
Given x =

 [1, 2, 3, 4]

,
????????
?       ?
?
?
?????????????>

Return false (not self crossing)
```

Example 3:

```
Given x =

 [1, 1, 1, 1]

,
?????
?   ?
?????>


Return true (self crossing)
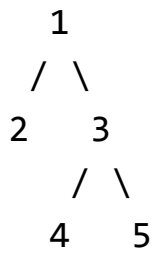```

Companies

Topics

Math

# 297. Serialize and Deserialize Binary Tree (Hard)

Serialization is the process of converting a data structure or object into a sequence of bits so that it can be stored in a file or memory buffer, or transmitted across a network connection link to be reconstructed later in the same or another computer environment.

Design an algorithm to serialize and deserialize a binary tree. There is no restriction on how your serialization/deserialization algorithm should work. You just need to ensure that a binary tree can be serialized to a string and this string can be deserialized to the original tree structure.

For example, you may serialize the following tree

```
    1
   / \
  2   3
     / \
    4   5
```

as `"[1,2,3,null,null,4,5]"`, just the same as how LeetCode OJ serializes a binary tree. You do not necessarily need to follow this format, so please be creative and come up with different approaches yourself.

Note: Do not use class member/global/static variables to store states. Your serialize and deserialize algorithms should be stateless.

Companies

Google Facebook Microsoft Amazon Bloomberg Uber LinkedIn Yahoo

Topics

Tree Design

73. Set Matrix Zeroes (Medium)

Given a m x n matrix, if an element is 0, set its entire row and column to 0. Do it in place.

Follow up:

Did you use extra space?

A straight forward solution using O(mn) space is probably a bad idea.

A simple improvement uses O(m + n) space, but still not the best solution.

Could you devise a constant space solution?

Companies

Microsoft   Amazon

Topics

Array

317. Shortest Distance from All Buildings (Hard)

You want to build a house on an empty land which reaches all buildings in the shortest amount of distance. You can only move up, down, left and right. You are given a 2D grid of values 0, 1 or 2, where:

- Each 0 marks an empty land which you can pass by freely.
- Each 1 marks a building which you cannot pass through.
- Each 2 marks an obstacle which you cannot pass through.

For example, given three buildings at `(0,0)`, `(0,4)`, `(2,2)`, and an obstacle at `(0,2)`:

```
1 - 0 - 2 - 0 - 1
|   |   |   |   |
0 - 0 - 0 - 0 - 0
|   |   |   |   |
0 - 0 - 1 - 0 - 0
```

The point `(1,2)` is an ideal empty land to build a house, as the total travel distance of 3+3+1=7 is minimal. So return 7.

Note:
There will be at least one building. If it is not possible to build such house according to the above rules, return -1.

Companies

Google   Zenefits

Topics

Breadth-first Search

214. Shortest Palindrome (Hard)

Given a string S, you are allowed to convert it to a palindrome by adding characters in front of it. Find and return the shortest palindrome you can find by performing this transformation.

For example:

Given `"aacecaaa"`, return `"aaacecaaa"`.

Given `"abcd"`, return `"dcbabcd"`.

Companies

Google   Pocket Gems

Topics

String

244. Shortest Word Distance II (Medium)

This is a follow up of Shortest Word Distance (/problems/shortest-word-distance). The only difference is now you are given the list of words and your method will be called repeatedly many times with different parameters. How would you optimize it?

Design a class which receives a list of words in the constructor, and implements a method that takes two words word1 and word2 and return the shortest distance between these two words in the list.

For example,
Assume that words = `["practice", "makes", "perfect", "coding", "makes"]`.

Given word1 = "`coding`", word2 = "`practice`", return 3.
Given word1 = "`makes`", word2 = "`coding`", return 1.

Note:
You may assume that word1 does not equal to word2, and word1 and word2 are both in the list.

Companies

(LinkedIn)

Topics

(Hash Table) (Design)

245. Shortest Word Distance III (Medium)

This is a follow up of Shortest Word Distance (/problems/shortest-word-distance). The only difference is now word1 could be the same as word2.

Given a list of words and two words word1 and word2, return the shortest distance between these two words in the list.

word1 and word2 may be the same and they represent two individual words in the list.

For example,
Assume that words = ["practice", "makes", "perfect", "coding", "makes"].

Given word1 = "makes", word2 = "coding", return 1.
Given word1 = "makes", word2 = "makes", return 3.

Note:
You may assume word1 and word2 are both in the list.

Companies

(LinkedIn)

Topics

(Array)

243. Shortest Word Distance (Easy)

Given a list of words and two words word1 and word2, return the shortest distance between these two words in the list.

For example,
Assume that words = `["practice", "makes", "perfect", "coding", "makes"]`.

Given word1 = "`coding`", word2 = "`practice`", return 3.
Given word1 = "`makes`", word2 = "`coding`", return 1.

Note:
You may assume that word1 does not equal to word2, and word1 and word2 are both in the list.

Companies

(LinkedIn)

Topics

(Array)

384. Shuffle an Array (Medium)

Shuffle a set of numbers without duplicates.

Example:

```
// Init an array with set 1, 2, and 3.
int[] nums = {1,2,3};
Solution solution = new Solution(nums);

// Shuffle the array [1,2,3] and return its result. Any permutation
 of [1,2,3] must equally likely to be returned.
solution.shuffle();

// Resets the array back to its original configuration [1,2,3].
solution.reset();

// Returns the random shuffling of array [1,2,3].
solution.shuffle();
```

Companies

Topics

71. Simplify Path (Medium)

Given an absolute path for a file (Unix-style), simplify it.

For example,
path = "/home/", => "/home"
path = "/a/./b/../../c/", => "/c"

Corner Cases:

- Did you consider the case where path = "/../" ?

  In this case, you should return "/" .

- Another corner case is the path might contain multiple slashes '/' together, such as "/home//foo/" .

  In this case, you should ignore redundant slashes and return "/home/foo" .

Companies

Facebook   Microsoft

Topics

String   Stack

137. Single Number II (Medium)

Given an array of integers, every element appears three times except for one, which appears exactly once. Find that single one.

Note:
Your algorithm should have a linear runtime complexity. Could you implement it without using extra memory?

Companies

Topics

Bit Manipulation

260. Single Number III (Medium)

Given an array of numbers `nums`, in which exactly two elements appear only once and all the other elements appear exactly twice. Find the two elements that appear only once.

For example:

Given `nums = [1, 2, 1, 3, 2, 5]`, return `[3, 5]`.

Note:

1. The order of the result is not important. So in the above example, `[5, 3]` is also correct.
2. Your algorithm should run in linear runtime complexity. Could you implement it using only constant space complexity?

Companies

Topics

Bit Manipulation

136. Single Number (Easy)

Given an array of integers, every element appears twice except for one. Find that single one.

Note:

Your algorithm should have a linear runtime complexity. Could you implement it without using extra memory?

Companies

Airbnb   Palantir

Topics

Hash Table   Bit Manipulation

239. Sliding Window Maximum (Hard)

Given an array nums, there is a sliding window of size k which is moving from the very left of the array to the very right. You can only see the k numbers in the window. Each time the sliding window moves right by one position.

For example,
Given nums = `[1,3,-1,-3,5,3,6,7]`, and k = 3.

```
Window position                 Max
---------------                 -----
[1   3  -1] -3   5   3   6   7     3
 1 [3  -1  -3] 5   3   6   7       3
 1   3 [-1  -3   5] 3   6   7      5
 1   3  -1 [-3   5   3] 6   7      5
 1   3  -1  -3 [5   3   6] 7       6
 1   3  -1  -3   5 [3   6   7]     7
```

Therefore, return the max sliding window as `[3,3,5,5,6,7]`.

Note:
You may assume k is always valid, ie: 1 ≤ k ≤ input array's size for non-empty array.

Follow up:
Could you solve it in linear time?

Companies

Google   Amazon   Zenefits

Topics

Heap

302. Smallest Rectangle Enclosing Black Pixels (Hard)

An image is represented by a binary matrix with `0` as a white pixel and `1` as a black pixel. The black pixels are connected, i.e., there is only one black region. Pixels are connected horizontally and vertically. Given the location `(x, y)` of one of the black pixels, return the area of the smallest (axis-aligned) rectangle that encloses all black pixels.

For example, given the following image:

```
[
  "0010",
  "0110",
  "0100"
]
```

and `x = 0`, `y = 2`,

Return `6`.

Companies

Google

Topics

Binary Search

75. Sort Colors (Medium)

Given an array with n objects colored red, white or blue, sort them so that objects of the same color are adjacent, with the colors in the order red, white and blue.

Here, we will use the integers 0, 1, and 2 to represent the color red, white, and blue respectively.

Note:
You are not suppose to use the library's sort function for this problem.

Follow up:

A rather straight forward solution is a two-pass algorithm using counting sort.

First, iterate the array counting number of 0's, 1's, and 2's, then overwrite array with total number of 0's, then 1's and followed by 2's.

Could you come up with an one-pass algorithm using only constant space?

Companies

Facebook   Microsoft   Pocket Gems

Topics

Array   Two Pointers   Sort

148. Sort List (Medium)

Sort a linked list in O(n log n) time using constant space complexity.

Companies

Topics

Linked List   Sort

360. Sort Transformed Array (Medium)

Given a sorted array of integers nums and integer values a, b and c. Apply a quadratic function of the form $f(x) = ax^2 + bx + c$ to each element x in the array.

The returned array must be in sorted order .

Expected time complexity: $O(n)$

Example:

```
nums = [-4, -2, 2, 4], a = 1, b = 3, c = 5,

Result: [3, 9, 15, 33]

nums = [-4, -2, 2, 4], a = -1, b = 3, c = 5

Result: [-23, -5, 1, 7]
```

Companies

Google

Topics

Math   Two Pointers

311. Sparse Matrix Multiplication (Medium)

Given two sparse matrices (https://en.wikipedia.org/wiki/Sparse_matrix) A and B, return the result of AB.

You may assume that A's column number is equal to B's row number.

Example:

```
A = [
  [ 1, 0, 0],
  [-1, 0, 3]
]

B = [
  [ 7, 0, 0 ],
  [ 0, 0, 0 ],
  [ 0, 0, 1 ]
]


       |   1 0 0 |     | 7 0 0 |     |  7 0 0 |
AB =   |  -1 0 3 |  x  | 0 0 0 |  =  | -7 0 3 |
                       | 0 0 1 |
```

Companies

Facebook   LinkedIn

Topics

Hash Table

59. Spiral Matrix II (Medium)

Given an integer n, generate a square matrix filled with elements from 1 to n^2 in spiral order.

For example,
Given n = 3,

You should return the following matrix:

```
[
  [ 1, 2, 3 ],
  [ 8, 9, 4 ],
  [ 7, 6, 5 ]
]
```

Companies

Topics

Array

54. Spiral Matrix (Medium)

Given a matrix of m x n elements (m rows, n columns), return all elements of the matrix in spiral order.

For example,
Given the following matrix:

```
[
  [ 1, 2, 3 ],
  [ 4, 5, 6 ],
  [ 7, 8, 9 ]
]
```

You should return `[1,2,3,6,9,8,7,4,5]`.

Companies

Google Microsoft Uber

Topics

Array

410. Split Array Largest Sum (Hard)

Given an array which consists of non-negative integers and an integer m, you can split the array into m non-empty continuous subarrays. Write an algorithm to minimize the largest sum among these m subarrays.

Note:
If n is the length of array, assume the following constraints are satisfied:

- 1 ≤ n ≤ 1000
- 1 ≤ m ≤ min(50, n)

Examples:

```
Input:
nums = [7,2,5,10,8]
m = 2

Output:
18

Explanation:
There are four ways to split nums into two subarrays.
The best way is to split it into [7,2,5] and [10,8],
where the largest sum among the two subarrays is only 18.
```

Companies

Facebook   Baidu

Topics

Binary Search   Dynamic Programming

69. Sqrt(x) (Easy)

Implement `int sqrt(int x)`.

Compute and return the square root of x.

x is guaranteed to be a non-negative integer.

Example 1:

```
Input: 4
Output: 2
```

Example 2:

```
Input: 8
Output: 2
Explanation: The square root of 8 is 2.82842..., and since we want t
o return an integer, the decimal part will be truncated.
```

Companies

Facebook   Bloomberg   Apple

Topics

Math   Binary Search

8. String to Integer (atoi) (Medium)

Implement `atoi` to convert a string to an integer.

Hint: Carefully consider all possible input cases. If you want a challenge, please do not see below and ask yourself what are the possible input cases.

Notes: It is intended for this problem to be specified vaguely (ie, no given input specs). You are responsible to gather all the input requirements up front.

Update (2015-02-10):
The signature of the `C++` function had been updated. If you still see your function signature accepts a `const char *` argument, please click the reload button ↻ to reset your code definition.

Requirements for atoi:

The function first discards as many whitespace characters as necessary until the first non-whitespace character is found.

Then, starting from this character, takes an optional initial plus or minus sign followed by as many numerical digits as possible, and interprets them as a numerical value.

The string can contain additional characters after those that form the integral number, which are ignored and have no effect on the behavior of this function.

If the first sequence of non-whitespace characters in str is not a valid integral number, or if no such sequence exists because either str is empty or it contains only whitespace characters, no conversion is performed.

If no valid conversion could be performed, a zero value is returned. If the correct value is out of the range of representable values, INT_MAX (2147483647) or INT_MIN (-2147483648) is returned.

Companies
(Microsoft) (Amazon) (Bloomberg) (Uber)

Topics
(Math) (String)

247. Strobogrammatic Number II (Medium)

A strobogrammatic number is a number that looks the same when rotated 180 degrees (looked at upside down).

Find all strobogrammatic numbers that are of length = n.

For example,
Given n = 2, return `["11","69","88","96"]`.

Companies

Google

Topics

Math Recursion

248. Strobogrammatic Number III (Hard)

A strobogrammatic number is a number that looks the same when rotated 180 degrees (looked at upside down).

Write a function to count the total strobogrammatic numbers that exist in the range of low <= num <= high.

For example,
Given low = "50", high = "100", return 3. Because 69, 88, and 96 are three strobogrammatic numbers.

Note:
Because the range might be a large number, the low and high numbers are represented as string.

Companies

Topics

(Math) (Recursion)

246. Strobogrammatic Number (Easy)

A strobogrammatic number is a number that looks the same when rotated 180 degrees (looked at upside down).

Write a function to determine if a number is strobogrammatic. The number is represented as a string.

For example, the numbers "69", "88", and "818" are all strobogrammatic.

Companies

Google

Topics

Hash Table   Math

90. Subsets II (Medium)

Given a collection of integers that might contain duplicates, nums , return all possible subsets (the power set).

Note: The solution set must not contain duplicate subsets.

For example,
If nums = `[1,2,2]` , a solution is:

```
[
  [2],
  [1],
  [1,2,2],
  [2,2],
  [1,2],
  []
]
```

Companies

Facebook

Topics

Array   Backtracking

78. Subsets (Medium)

Given a set of distinct integers, nums, return all possible subsets (the power set).

Note: The solution set must not contain duplicate subsets.

For example,
If nums = `[1,2,3]`, a solution is:

```
[
  [3],
  [1],
  [2],
  [1,2,3],
  [1,3],
  [2,3],
  [1,2],
  []
]
```

Companies

Facebook   Amazon   Bloomberg   Uber   Coupang

Topics

Array   Backtracking   Bit Manipulation

30. Substring with Concatenation of All Words (Hard)

You are given a string, s, and a list of words, words , that are all of the same length. Find all starting indices of substring(s) in s that is a concatenation of each word in words exactly once and without any intervening characters.

For example, given:
s: `"barfoothefoobarman"`
words : `["foo", "bar"]`

You should return the indices: `[0,9]`.
(order does not matter).

Companies

Topics

(Hash Table) (Two Pointers) (String)

37. Sudoku Solver (Hard)

Write a program to solve a Sudoku puzzle by filling the empty cells.

Empty cells are indicated by the character '.'.

You may assume that there will be only one unique solution.

5	3			7				
6			1	9	5			
	9	8					6	
8				6				3
4			8		3			1
7				2				6
	6					2	8	
			4	1	9			5
				8			7	9

A sudoku puzzle...

5	3	4	6	7	8	9	1	2
6	7	2	1	9	5	3	4	8
1	9	8	3	4	2	5	6	7
8	5	9	7	6	1	4	2	3
4	2	6	8	5	3	7	9	1
7	1	3	9	2	4	8	5	6
9	6	1	5	3	7	2	8	4
2	8	7	4	1	9	6	3	5
3	4	5	2	8	6	1	7	9

...and its solution numbers marked in red.

Companies

Uber   Snapchat

Topics

Hash Table   Backtracking

404. Sum of Left Leaves (Easy)

Find the sum of all left leaves in a given binary tree.

Example:

```
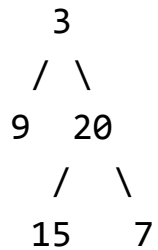   3
  / \
 9  20
   /  \
  15   7
```

There are two left leaves in the binary tree, with values 9 and 15 respectively. Return 24.

Companies

Facebook

Topics

Tree

371. Sum of Two Integers (Easy)

Calculate the sum of two integers a and b, but you are not allowed   to use the operator
 `+` and `-` .

Example:
Given a = 1 and b = 2, return 3.

Companies

Hulu

Topics

Bit Manipulation

129. Sum Root to Leaf Numbers (Medium)

Given a binary tree containing digits from `0-9` only, each root-to-leaf path could represent a number.

An example is the root-to-leaf path `1->2->3` which represents the number `123`.

Find the total sum of all root-to-leaf numbers.

For example,

```
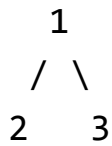   1
  / \
 2   3
```

The root-to-leaf path `1->2` represents the number `12`.
The root-to-leaf path `1->3` represents the number `13`.

Return the sum = 12 + 13 = `25`.

Companies

Topics

Tree   Depth-first Search

228. Summary Ranges (Medium)

Given a sorted integer array without duplicates, return the summary of its ranges.

Example 1:

```
Input: [0,1,2,4,5,7]
Output: ["0->2","4->5","7"]
```

Example 2:

```
Input: [0,2,3,4,6,8,9]
Output: ["0","2->4","6","8->9"]
```

Companies

Google

Topics

Array

372. Super Pow (Medium)

Your task is to calculate a^b mod 1337 where a is a positive integer and b is an extremely large positive integer given in the form of an array.

Example1:

```
a = 2
b = [3]

Result: 8
```

Example2:

```
a = 2
b = [1,0]

Result: 1024
```

Companies

Topics

Math

313. Super Ugly Number (Medium)

Write a program to find the nth super ugly number.

Super ugly numbers are positive numbers whose all prime factors are in the given prime list `primes` of size `k`. For example, `[1, 2, 4, 7, 8, 13, 14, 16, 19, 26, 28, 32]` is the sequence of the first 12 super ugly numbers given `primes = [2, 7, 13, 19]` of size 4.

Note:
(1) `1` is a super ugly number for any given `primes`.
(2) The given numbers in `primes` are in ascending order.
(3) $0 < k \leq 100$, $0 < n \leq 10^6$, $0 < $ `primes[i]` $ < 1000$.
(4) The nth super ugly number is guaranteed to fit in a 32-bit signed integer.

Companies

Google

Topics

Math Heap

130. Surrounded Regions (Medium)

Given a 2D board containing `'X'` and `'O'` (the letter O), capture all regions surrounded by `'X'`.

A region is captured by flipping all `'O'`s into `'X'`s in that surrounded region.

For example,

```
X X X X
X O O X
X X O X
X O X X
```

After running your function, the board should be:

```
X X X X
X X X X
X X X X
X O X X
```

Companies

Topics

Breadth-first Search   Union Find

24. Swap Nodes in Pairs (Medium)

Given a linked list, swap every two adjacent nodes and return its head.

For example,
Given `1->2->3->4`, you should return the list as `2->1->4->3`.

Your algorithm should use only constant space. You may not modify the values in the list, only nodes itself can be changed.

Companies

Microsoft   Bloomberg   Uber

Topics

Linked List

101. Symmetric Tree (Easy)

Given a binary tree, check whether it is a mirror of itself (ie, symmetric around its center).

For example, this binary tree `[1,2,2,3,4,4,3]` is symmetric:

```
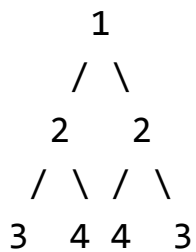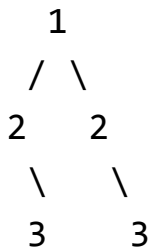    1
   / \
  2   2
 / \ / \
3  4 4  3
```

But the following `[1,2,2,null,3,null,3]` is not:

```
    1
   / \
  2   2
   \   \
   3   3
```

Note:
Bonus points if you could solve it both recursively and iteratively.

Companies

(Microsoft) (Bloomberg) (LinkedIn)

Topics

(Tree) (Depth-first Search) (Breadth-first Search)

68. Text Justification (Hard)

Given an array of words and a length L, format the text such that each line has exactly L characters and is fully (left and right) justified.

You should pack your words in a greedy approach; that is, pack as many words as you can in each line. Pad extra spaces `' '` when necessary so that each line has exactly L characters.

Extra spaces between words should be distributed as evenly as possible. If the number of spaces on a line do not divide evenly between words, the empty slots on the left will be assigned more spaces than the slots on the right.

For the last line of text, it should be left justified and no extra space is inserted between words.

For example,
words : ["This", "is", "an", "example", "of", "text", "justification."]
L: 16.

Return the formatted lines as:

```
[
   "This    is    an",
   "example  of text",
   "justification.  "
]
```

Note: Each word is guaranteed not to exceed L in length.

Corner Cases:

- A line other than the last line might contain only one word. What should you do in this case?

  In this case, that line should be left-justified.

Companies

Facebook LinkedIn Airbnb

Topics

String

218. The Skyline Problem (Hard)

A city's skyline is the outer contour of the silhouette formed by all the buildings in that city when viewed from a distance. Now suppose you are given the locations and height of all the buildings   as shown on a cityscape photo (Figure A), write a program to output the skyline   formed by these buildings collectively (Figure B).

```
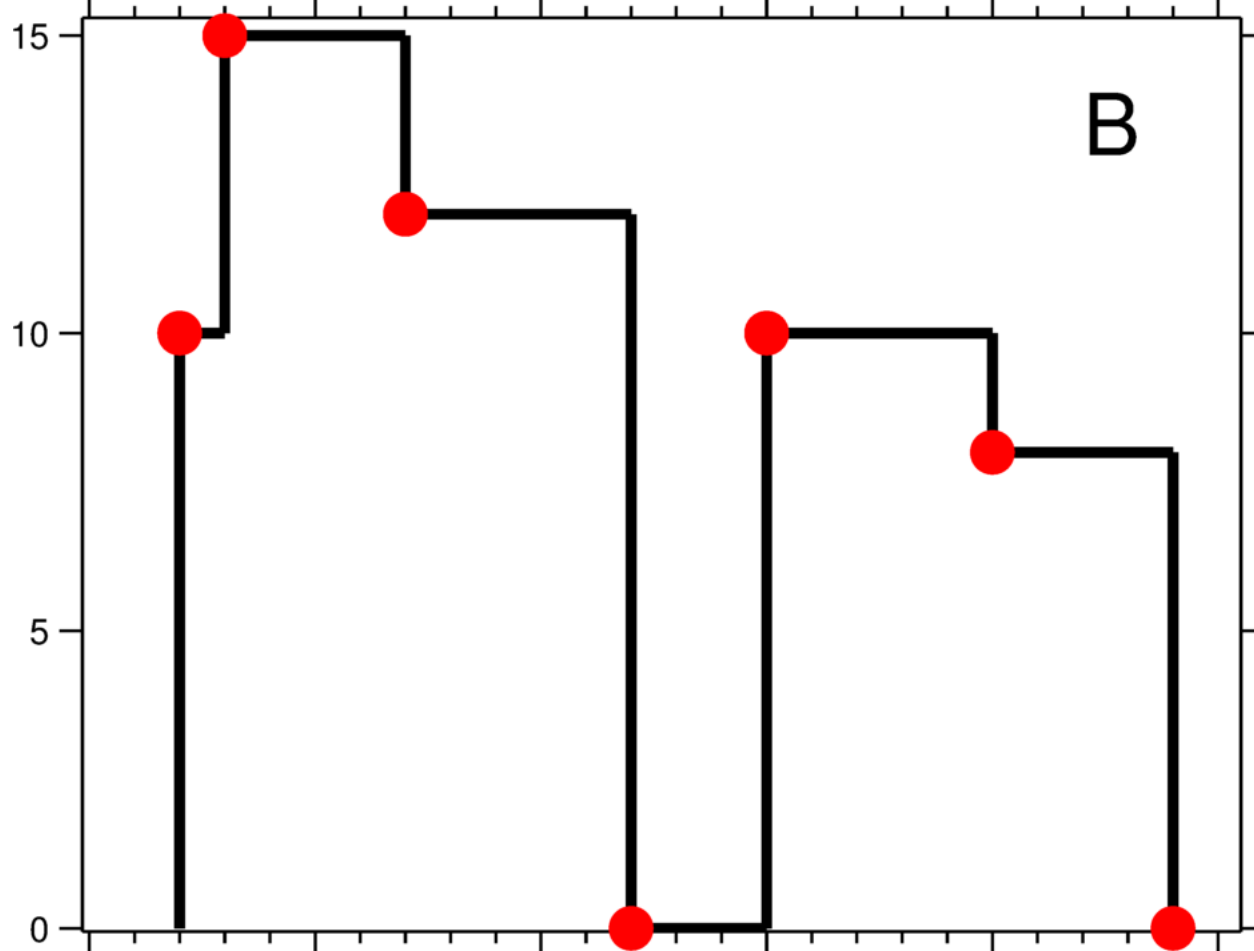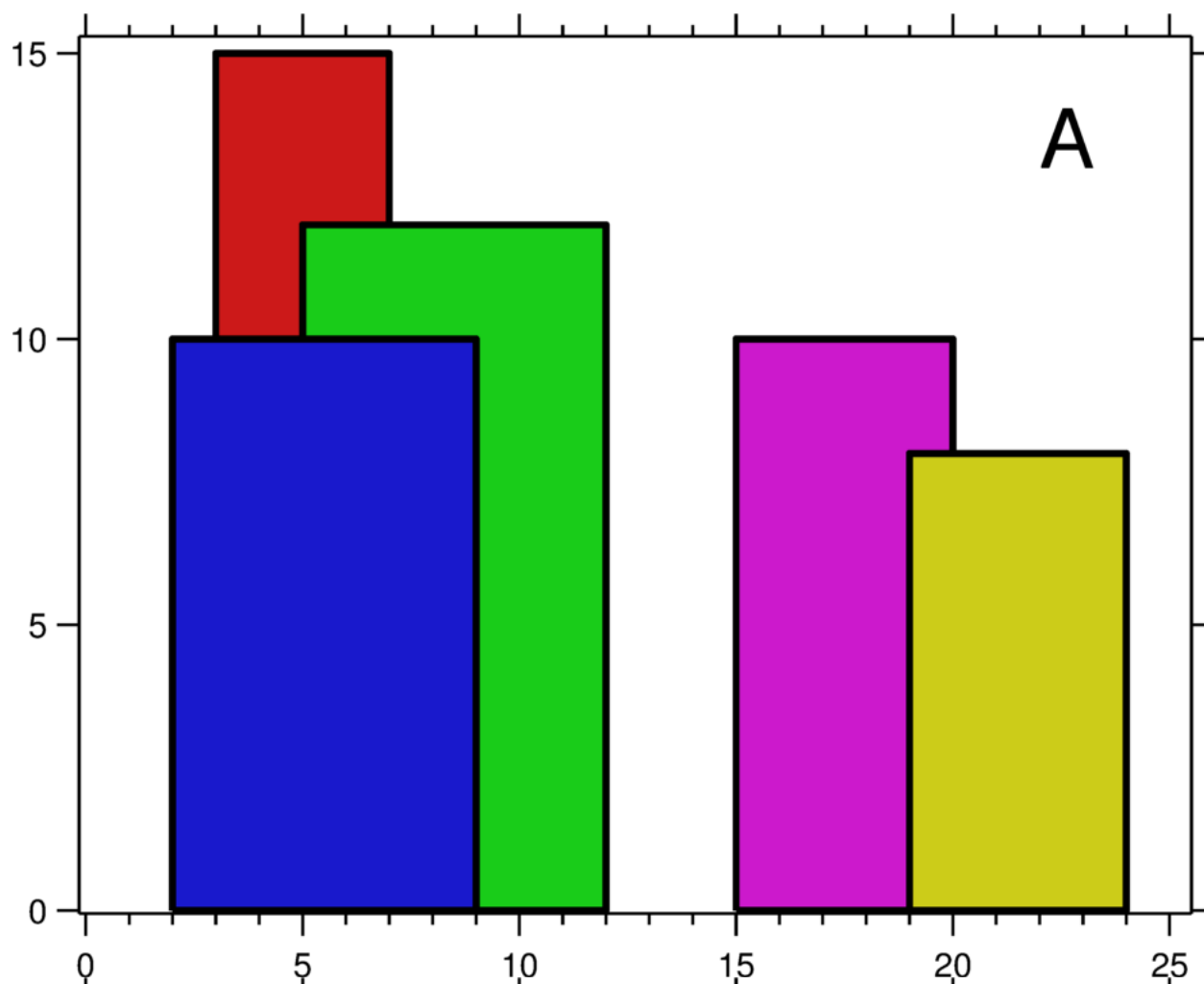  .          .            .              .             .              . .
  0          5           10             15            20             25
```

The geometric information of each building is represented by a triplet of integers `[Li, Ri, Hi]`, where `Li` and `Ri` are the x coordinates of the left and right edge of the ith building, respectively, and `Hi` is its height. It is guaranteed that `0 ≤ Li, Ri ≤ INT_MAX`, `0 < Hi ≤ INT_MAX`, and `Ri - Li > 0`. You may assume all buildings are perfect rectangles grounded on an absolutely flat surface at height 0.

For instance, the dimensions of all buildings in Figure A are recorded as: `[ [2 9 10], [3 7 15], [5 12 12], [15 20 10], [19 24 8] ]` .

The output is a list of "key points " (red dots in Figure B) in the format of `[ [x1,y1], [x2, y2], [x3, y3], ... ]` that uniquely defines a skyline. A key point is the left endpoint of a horizontal line segment . Note that the last key point, where the rightmost building ends, is merely used to mark the termination of the skyline, and always has zero height. Also, the ground in between any two adjacent buildings should be considered part of the skyline contour.

For instance, the skyline in Figure B should be represented as: `[ [2 10], [3 15], [7 12], [12 0], [15 10], [20 8], [24, 0] ]`.

Notes:

- The number of buildings in any input list is guaranteed to be in the range `[0, 10000]`.
- The input list is already sorted in ascending order by the left x position `Li`.
- The output list must be sorted by the x position.
- There must be no consecutive horizontal lines of equal height in the output skyline. For instance, `[...[2 3], [4 5], [7 5], [11 5], [12 7]...]` is not acceptable; the three lines of height 5 should be merged into one in the final output as such: `[...[2 3], [4 5], [12 7], ...]`

Companies

Google   Facebook   Microsoft   Twitter   Yelp

Topics

Divide and Conquer   Heap   Binary Indexed Tree   Segment Tree

414. Third Maximum Number (Easy)

Given a non-empty array of integers, return the third maximum number in this array. If it does not exist, return the maximum number. The time complexity must be in O(n).

Example 1:

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

Output: 1

Explanation: The third maximum is 1.
```

Example 2:

```
Input: [1, 2]

Output: 2

Explanation: The third maximum does not exist, so the maximum (2) is
  returned instead.
```

Example 3:

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

Output: 1

Explanation: Note that the third maximum here means the third maximu
m distinct number.
Both numbers with value 2 are both considered as second maximum.
```

Companies

Amazon

Topics

Array

347. Top K Frequent Elements (Medium)

Given a non-empty array of integers, return the k most frequent elements.

For example,
Given `[1,1,1,2,2,3]` and k = 2, return `[1,2]`.

Note:

- You may assume k is always valid, 1 ≤ k ≤ number of unique elements.
- Your algorithm's time complexity must be better than O(n log n), where n is the array's size.

Companies

Yelp   Pocket Gems

Topics

Hash Table   Heap

407. Trapping Rain Water II (Hard)

Given an `m x n` matrix of positive integers representing the height of each unit cell in a 2D elevation map, compute the volume of water it is able to trap after raining.

Note:
Both m and n are less than 110. The height of each unit cell is greater than 0 and is less than 20,000.

Example:

```
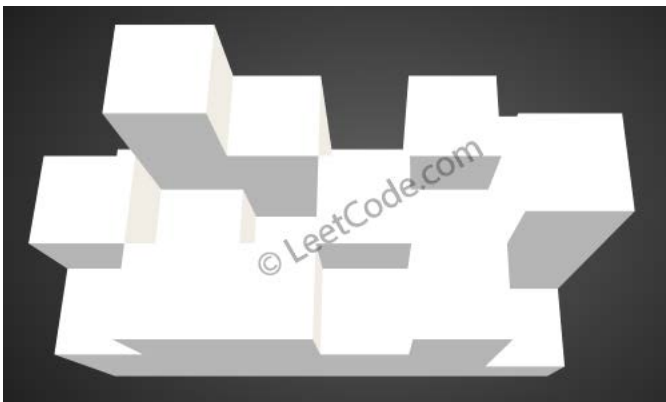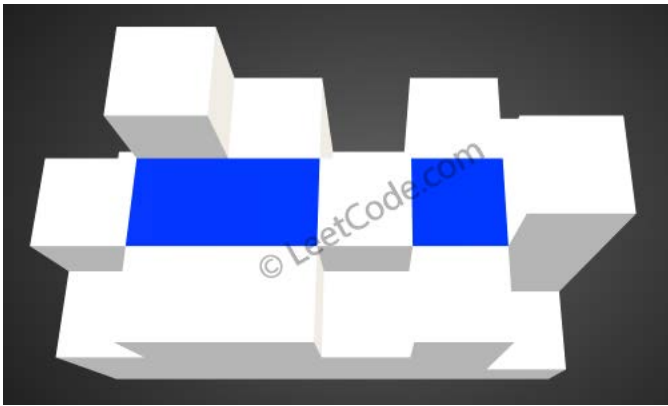Given the following 3x6 height map:
[
  [1,4,3,1,3,2],
  [3,2,1,3,2,4],
  [2,3,3,2,3,1]
]

Return 4.
```


The above image represents the elevation map `[[1,4,3,1,3,2],[3,2,1,3,2,4], [2,3,3,2,3,1]]` before the rain.

After the rain, water are trapped between the blocks. The total volume of water trapped is 4.

Companies

Google   Twitter

Topics

Heap   Breadth-first Search

42. Trapping Rain Water (Hard)

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

For example,
Given `[0,1,0,2,1,0,1,3,2,1,2,1]`, return `6`.

The above elevation map is represented by array [0,1,0,2,1,0,1,3,2,1,2,1]. In this case, 6 units of rain water (blue section) are being trapped. Thanks Marcos for contributing this image!

Companies

Google   Amazon   Bloomberg   Twitter   Apple   Zenefits

Topics

Array   Two Pointers   Stack

120. Triangle (Medium)

Given a triangle, find the minimum path sum from top to bottom. Each step you may move to adjacent numbers on the row below.

For example, given the following triangle

```
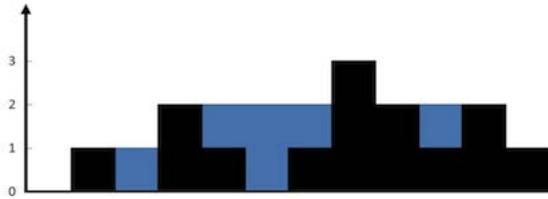[
     [2],
    [3,4],
   [6,5,7],
  [4,1,8,3]
]
```

The minimum path sum from top to bottom is `11` (i.e., 2 + 3 + 5 + 1 = 11).

Note:
Bonus point if you are able to do this using only O(n) extra space, where n is the total number of rows in the triangle.

Companies

Topics

Array   Dynamic Programming

167. Two Sum II - Input array is sorted (Easy)

Given an array of integers that is already sorted in ascending order , find two numbers such that they add up to a specific target number.

The function twoSum should return indices of the two numbers such that they add up to the target, where index1 must be less than index2. Please note that your returned answers (both index1 and index2) are not zero-based.

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

```
Input: numbers={2, 7, 11, 15}, target=9
Output: index1=1, index2=2
```

Companies

Amazon

Topics

Array   Two Pointers   Binary Search

170. Two Sum III - Data structure design (Easy)

Design and implement a TwoSum class. It should support the following operations: `add` and `find`.

`add` - Add the number to an internal data structure.
`find` - Find if there exists any pair of numbers which sum is equal to the value.

For example,

```
add(1); add(3); add(5);
find(4) -> true
find(7) -> false
```

Companies

LinkedIn

Topics

Hash Table   Design

1. Two Sum (Easy)

Given an array of integers, return indices of the two numbers such that they add up to a specific target.

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

Example:

```
Given nums = [2, 7, 11, 15], target = 9,

Because nums[0] + nums[1] = 2 + 7 = 9,
return [0, 1].
```

Companies

Facebook   Microsoft   Amazon   Bloomberg   Uber   LinkedIn   Apple   Airbnb   Yelp   Yahoo

Adobe   Dropbox

Topics

Array   Hash Table

264. Ugly Number II (Medium)

Write a program to find the `n`-th ugly number.

Ugly numbers are positive numbers whose prime factors only include `2`, `3`, `5`. For example, `1, 2, 3, 4, 5, 6, 8, 9, 10, 12` is the sequence of the first `10` ugly numbers.

Note that `1` is typically treated as an ugly number, and n does not exceed 1690 .

Companies

Topics

Math   Dynamic Programming   Heap

263. Ugly Number (Easy)

Write a program to check whether a given number is an ugly number.

Ugly numbers are positive numbers whose prime factors only include `2`, `3`, `5`. For example, `6`, `8` are ugly while `14` is not ugly since it includes another prime factor `7`.

Note that `1` is typically treated as an ugly number.

Companies

Topics

Math

95. Unique Binary Search Trees II (Medium)

Given an integer n, generate all structurally unique BST's (binary search trees) that store values 1...n.

For example,
Given n = 3, your program should return all 5 unique BST's shown below.

```
   1         3     3      2      1
    \       /     /      / \      \
     3     2     1      1   3      2
    /     /       \                 \
   2     1         2                 3
```

Companies

Topics

Dynamic Programming   Tree

96. Unique Binary Search Trees (Medium)

Given n, how many structurally unique BST's (binary search trees) that store values 1...n?

For example,
Given n = 3, there are a total of 5 unique BST's.

```
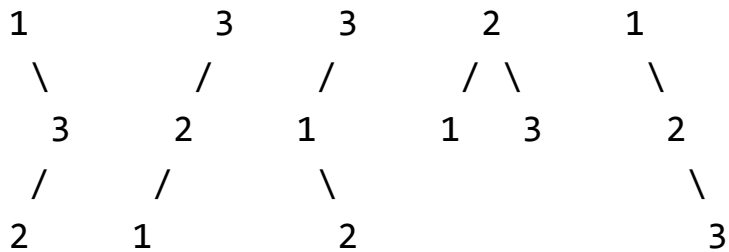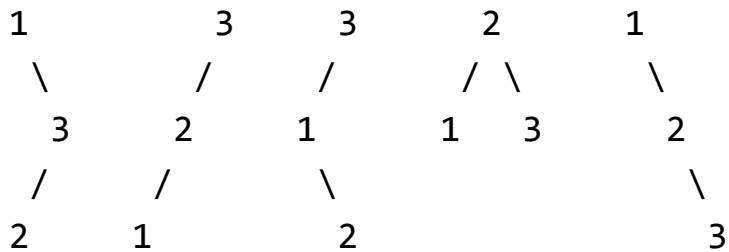   1           3     3       2        1
    \         /     /       / \        \
     3       2     1       1   3        2
    /       /       \                    \
   2       1         2                    3
```

Companies

Snapchat

Topics

Dynamic Programming   Tree

63. Unique Paths II (Medium)

Follow up for "Unique Paths":

Now consider if some obstacles are added to the grids. How many unique paths would there be?

An obstacle and empty space is marked as `1` and `0` respectively in the grid.

For example,

There is one obstacle in the middle of a 3x3 grid as illustrated below.

```
[
  [0,0,0],
  [0,1,0],
  [0,0,0]
]
```

The total number of unique paths is `2`.

Note: m and n will be at most 100.

Companies

Bloomberg

Topics

Array   Dynamic Programming

62. Unique Paths (Medium)

A robot is located at the top-left corner of a m x n grid (marked 'Start' in the diagram below).

The robot can only move either down or right at any point in time. The robot is trying to reach the bottom-right corner of the grid (marked 'Finish' in the diagram below).

How many possible unique paths are there?

Above is a 3 x 7 grid. How many possible unique paths are there?

Note: m and n will be at most 100.

Companies

Bloomberg

Topics

Array   Dynamic Programming

288. Unique Word Abbreviation (Medium)

An abbreviation of a word follows the form <first letter><number><last letter>. Below are some examples of word abbreviations:

```
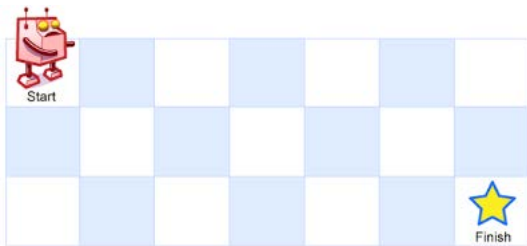a) it                       --> it    (no abbreviation)


     1
b) d|o|g                    --> d1g


            1    1 1
    1---5----0----5--8
c) i|nternationalizatio|n   --> i18n


            1
    1---5----0
d) l|ocalizatio|n           --> l10n
```

Assume you have a dictionary and given a word, find whether its abbreviation is unique in the dictionary. A word's abbreviation is unique if no other word from the dictionary has the same abbreviation.

Example:

```
Given dictionary = [ "deer", "door", "cake", "card" ]

isUnique("dear") ->
 false

isUnique("cart") ->
 true

isUnique("cane") ->
 false

isUnique("make") ->
 true
```

Companies

Google

Topics

Hash Table   Design

393. UTF-8 Validation (Medium)

A character in UTF8 can be from 1 to 4 bytes long, subjected to the following rules:

1. For 1-byte character, the first bit is a 0, followed by its unicode code.
2. For n-bytes character, the first n-bits are all one's, the n+1 bit is 0, followed by n-1 bytes with most significant 2 bits being 10.

This is how the UTF-8 encoding would work:

```
   Char. number range  |        UTF-8 octet sequence
      (hexadecimal)     |              (binary)
   -------------------+-------------------------------------
 ---
   0000 0000-0000 007F | 0xxxxxxx
   0000 0080-0000 07FF | 110xxxxx 10xxxxxx
   0000 0800-0000 FFFF | 1110xxxx 10xxxxxx 10xxxxxx
   0001 0000-0010 FFFF | 11110xxx 10xxxxxx 10xxxxxx 10xxxxxx
```

Given an array of integers representing the data, return whether it is a valid utf-8 encoding.

Note:
The input is an array of integers. Only the least significant 8 bits   of each integer is used to store the data. This means each integer represents only 1 byte of data.

Example 1:

```
data = [197, 130, 1], which represents the octet sequence: 11000101
 10000010 00000001.

Return true.
It is a valid utf-8 encoding for a 2-bytes character followed by a 1
-byte character.
```

Example 2:

data = [235, 140, 4], which represented the octet sequence: 11101011 10001100 00000100.

Return false.
The first 3 bits are all one's and the 4th bit is 0 means it is a 3-bytes character.
The next byte is a continuation byte which starts with 10 and that's correct.
But the second continuation byte does not start with 10, so it is in valid.

Companies

Google

Topics

Bit Manipulation

242. Valid Anagram (Easy)

Given two strings s and t, write a function to determine if t is an anagram of s.

For example,

s = "anagram", t = "nagaram", return true.

s = "rat", t = "car", return false.

Note:

You may assume the string contains only lowercase alphabets.

Follow up:

What if the inputs contain unicode characters? How would you adapt your solution to such case?

Companies

Amazon   Uber   Yelp

Topics

Hash Table   Sort

65. Valid Number (Hard)

Validate if a given string is numeric.

Some examples:

`"0"` => `true`

`" 0.1 "` => `true`

`"abc"` => `false`

`"1 a"` => `false`

`"2e10"` => `true`

Note: It is intended for the problem statement to be ambiguous. You should gather all requirements up front before implementing one.

Update (2015-02-10):
The signature of the `C++` function had been updated. If you still see your function signature accepts a `const char *` argument, please click the reload button ↻ to reset your code definition.

Companies

(LinkedIn)

Topics

(Math) (String)

125. Valid Palindrome (Easy)

Given a string, determine if it is a palindrome, considering only alphanumeric characters and ignoring cases.

For example,
`"A man, a plan, a canal: Panama"` is a palindrome.
`"race a car"` is not a palindrome.

Note:
Have you consider that the string might be empty? This is a good question to ask during an interview.

For the purpose of this problem, we define empty string as valid palindrome.

Companies

Facebook   Microsoft   Uber   Zenefits

Topics

Two Pointers   String

20. Valid Parentheses (Easy)

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

The brackets must close in the correct order, `"()"` and `"()[]{}"` are all valid but `"(]"` and `"([)]"` are not.

Companies

Google Facebook Microsoft Amazon Bloomberg Twitter Airbnb Zenefits

Topics

String Stack

367. Valid Perfect Square (Easy)

Given a positive integer num, write a function which returns True if num is a perfect square else False.

Note: Do not use any built-in library function such as `sqrt`.

Example 1:

```
Input: 16
Returns: True
```

Example 2:

```
Input: 14
Returns: False
```

Companies

LinkedIn

Topics

Math   Binary Search

36. Valid Sudoku (Medium)

Determine if a Sudoku is valid, according to: Sudoku Puzzles - The Rules (http://sudoku.com.au/TheRules.aspx).

The Sudoku board could be partially filled, where empty cells are filled with the character `.`.

5	3			7				
6			1	9	5			
	9	8					6	
8				6				3
4			8		3			1
7				2				6
	6					2	8	
			4	1	9			5
				8			7	9

A partially filled sudoku which is valid.

Note:
A valid Sudoku board (partially filled) is not necessarily solvable. Only the filled cells need to be validated.

Companies

Uber Apple Snapchat

Topics

Hash Table

408. Valid Word Abbreviation (Easy)

Given a non-empty string `s` and an abbreviation `abbr`, return whether the string matches with the given abbreviation.

A string such as `"word"` contains only the following valid abbreviations:

```
["word", "1ord", "w1rd", "wo1d", "wor1", "2rd", "w2d", "wo2", "1o1
d", "1or1", "w1r1", "1o2", "2r1", "3d", "w3", "4"]
```

Notice that only the above abbreviations are valid abbreviations of the string `"word"`. Any other string is not a valid abbreviation of `"word"`.

Note:
Assume `s` contains only lowercase letters and `abbr` contains only lowercase letters and digits.

Example 1:

```
Given s = "internationalization", abbr = "i12iz4n":


Return true.
```

Example 2:

```
Given s = "apple", abbr = "a2e":


Return false.
```

Companies

Google

Topics

String

98. Validate Binary Search Tree (Medium)

Given a binary tree, determine if it is a valid binary search tree (BST).

Assume a BST is defined as follows:

- The left subtree of a node contains only nodes with keys less than the node's key.
- The right subtree of a node contains only nodes with keys greater than the node's key.
- Both the left and right subtrees must also be binary search trees.

Example 1:

```
    2
   / \
  1   3
```

Binary tree `[2,1,3]`, return true.

Example 2:

```
    1
   / \
  2   3
```

Binary tree `[1,2,3]`, return false.

Companies

Facebook Microsoft Amazon Bloomberg

Topics

Tree Depth-first Search

255. Verify Preorder Sequence in Binary Search Tree (Medium)

Given an array of numbers, verify whether it is the correct preorder traversal sequence of a binary search tree.

You may assume each number in the sequence is unique.

Follow up:
Could you do it using only constant space complexity?

Companies

Zenefits

Topics

Stack   Tree

331. Verify Preorder Serialization of a Binary Tree (Medium)

One way to serialize a binary tree is to use pre-order traversal. When we encounter a non-null node, we record the node's value. If it is a null node, we record using a sentinel value such as `#`.

```
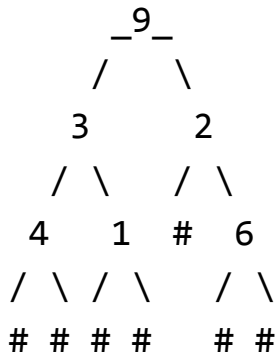     _9_
    /   \
   3     2
  / \   / \
 4   1 #   6
/ \ / \   / \
# # # #   # #
```

For example, the above binary tree can be serialized to the string `"9,3,4,#,#,1,#,#,2,#,6,#,#"`, where `#` represents a null node.

Given a string of comma separated values, verify whether it is a correct preorder traversal serialization of a binary tree. Find an algorithm without reconstructing the tree.

Each comma separated value in the string must be either an integer or a character `'#'` representing `null` pointer.

You may assume that the input format is always valid, for example it could never contain two consecutive commas such as `"1,,3"`.

Example 1:
`"9,3,4,#,#,1,#,#,2,#,6,#,#"`
Return `true`

Example 2:
`"1,#"`
Return `false`

Example 3:

`"9,#,#,1"`

Return `false`

Companies

Google

Topics

Stack

286. Walls and Gates (Medium)

You are given a m x n 2D grid initialized with these three possible values.

1. `-1` - A wall or an obstacle.
2. `0` - A gate.
3. `INF` - Infinity means an empty room. We use the value 2^{31} - `1` = `2147483647` to represent `INF` as you may assume that the distance to a gate is less than `2147483647`.

Fill each empty room with the distance to its nearest gate. If it is impossible to reach a gate, it should be filled with `INF`.

For example, given the 2D grid:

```
INF  -1   0   INF
INF INF INF   -1
INF  -1 INF   -1
  0  -1 INF INF
```

After running your function, the 2D grid should be:

```
  3  -1   0    1
  2   2   1   -1
  1  -1   2   -1
  0  -1   3    4
```

Companies

Google   Facebook

Topics

Breadth-first Search

365. Water and Jug Problem (Medium)

You are given two jugs with capacities x and y litres. There is an infinite amount of water supply available. You need to determine whether it is possible to measure exactly z litres using these two jugs.

If z liters of water is measurable, you must have z liters of water contained within one or both buckets   by the end.

Operations allowed:

- Fill any of the jugs completely with water.
- Empty any of the jugs.
- Pour water from one jug into another till the other jug is completely full or the first jug itself is empty.

Example 1: (From the famous "Die Hard" example (https://www.youtube.com/watch?v=BVtQNK_ZUJg))

```
Input: x = 3, y = 5, z = 4
Output: True
```

Example 2:

```
Input: x = 2, y = 6, z = 5
Output: False
```

Companies

Microsoft

Topics

Math

324. Wiggle Sort II (Medium)

Given an unsorted array `nums`, reorder it such that `nums[0]` < `nums[1]` > `nums[2]` < `nums[3]`....

Example:

(1) Given `nums = [1, 5, 1, 1, 6, 4]`, one possible answer is `[1, 4, 1, 5, 1, 6]`.

(2) Given `nums = [1, 3, 2, 2, 3, 1]`, one possible answer is `[2, 3, 1, 3, 1, 2]`.

Note:

You may assume all input has valid answer.

Follow Up:

Can you do it in O(n) time and/or in-place with O(1) extra space?

Companies

Google

Topics

Sort

280. Wiggle Sort (Medium)

Given an unsorted array `nums`, reorder it in-place such that `nums[0] <= nums[1] >= nums[2] <= nums[3]....`

For example, given `nums = [3, 5, 2, 1, 6, 4]`, one possible answer is `[1, 6, 2, 5, 3, 4]`.

Companies

Google

Topics

Array   Sort

376. Wiggle Subsequence (Medium)

A sequence of numbers is called a wiggle sequence if the differences between successive numbers strictly alternate between positive and negative. The first difference (if one exists) may be either positive or negative. A sequence with fewer than two elements is trivially a wiggle sequence.

For example, `[1,7,4,9,2,5]` is a wiggle sequence because the differences (6,-3,5,-7,3) are alternately positive and negative. In contrast, `[1,4,7,2,5]` and `[1,7,4,5,5]` are not wiggle sequences, the first because its first two differences are positive and the second because its last difference is zero.

Given a sequence of integers, return the length of the longest subsequence that is a wiggle sequence. A subsequence is obtained by deleting some number of elements (eventually, also zero) from the original sequence, leaving the remaining elements in their original order.

Examples:

```
Input: [1,7,4,9,2,5]
Output: 6
The entire sequence is a wiggle sequence.

Input: [1,17,5,10,13,15,10,5,16,8]
Output: 7
There are several subsequences that achieve this length. One is [1,1
7,10,13,10,16,8].

Input: [1,2,3,4,5,6,7,8,9]
Output: 2
```

Follow up:
Can you do it in O(n) time?

Companies

Topics

Dynamic Programming   Greedy

44. Wildcard Matching (Hard)

Implement wildcard pattern matching with support for '?' and '*'.

```
'?' Matches any single character.
'*' Matches any sequence of characters (including the empty sequenc
e).

The matching should cover the entire input string (not partial).

The function prototype should be:
bool isMatch(const char *s, const char *p)

Some examples:
isMatch("aa","a") → false
isMatch("aa","aa") → true
isMatch("aaa","aa") → false
isMatch("aa", "*") → true
isMatch("aa", "a*") → true
isMatch("ab", "?*") → true
isMatch("aab", "c*a*b") → false
```

Companies

Google   Facebook   Twitter   Snapchat   Two Sigma

Topics

String   Dynamic Programming   Backtracking   Greedy

140. Word Break II (Hard)

Given a non-empty string s and a dictionary wordDict containing a list of non-empty words, add spaces in s to construct a sentence where each word is a valid dictionary word. You may assume the dictionary does not contain duplicate words.

Return all such possible sentences.

For example, given
s = `"catsanddog"`,
dict = `["cat", "cats", "and", "sand", "dog"]`.

A solution is `["cats and dog", "cat sand dog"]`.

UPDATE (2017/1/4):
The wordDict parameter had been changed to a list of strings (instead of a set of strings). Please reload the code definition to get the latest changes.

Companies

Google   Uber   Twitter   Snapchat   Dropbox

Topics

Dynamic Programming   Backtracking

139. Word Break (Medium)

Given a non-empty string s and a dictionary wordDict containing a list of non-empty words, determine if s can be segmented into a space-separated sequence of one or more dictionary words. You may assume the dictionary does not contain duplicate words.

For example, given
s = `"leetcode"`,
dict = `["leet", "code"]`.

Return true because `"leetcode"` can be segmented as `"leet code"`.

UPDATE (2017/1/4):
The wordDict parameter had been changed to a list of strings (instead of a set of strings). Please reload the code definition to get the latest changes.

Companies

Google   Facebook   Amazon   Bloomberg   Uber   Yahoo   Pocket Gems   Square   Coupang

Topics

Dynamic Programming

126. Word Ladder II (Hard)

Given two words (beginWord and endWord), and a dictionary's word list, find all shortest transformation sequence(s) from beginWord to endWord, such that:

1. Only one letter can be changed at a time
2. Each transformed word must exist in the word list. Note that beginWord is not a transformed word.

For example,

Given:
beginWord = `"hit"`
endWord = `"cog"`
wordList = `["hot","dot","dog","lot","log","cog"]`

Return

```
[
  ["hit","hot","dot","dog","cog"],
  ["hit","hot","lot","log","cog"]
]
```

Note:

- Return an empty list if there is no such transformation sequence.
- All words have the same length.
- All words contain only lowercase alphabetic characters.
- You may assume no duplicates in the word list.
- You may assume beginWord and endWord are non-empty and are not the same.

UPDATE (2017/1/20):
The wordList parameter had been changed to a list of strings (instead of a set of strings). Please reload the code definition to get the latest changes.

Companies

Amazon   Yelp

Topics

Array String Backtracking Breadth-first Search

127. Word Ladder (Medium)

Given two words (beginWord and endWord), and a dictionary's word list, find the length of shortest transformation sequence from beginWord to endWord, such that:

1. Only one letter can be changed at a time.
2. Each transformed word must exist in the word list. Note that beginWord is not a transformed word.

For example,

Given:
beginWord = `"hit"`
endWord = `"cog"`
wordList = `["hot","dot","dog","lot","log","cog"]`

As one shortest transformation is `"hit" -> "hot" -> "dot" -> "dog" -> "cog"`, return its length `5`.

Note:

- Return 0 if there is no such transformation sequence.
- All words have the same length.
- All words contain only lowercase alphabetic characters.
- You may assume no duplicates in the word list.
- You may assume beginWord and endWord are non-empty and are not the same.

UPDATE (2017/1/20):
The wordList parameter had been changed to a list of strings (instead of a set of strings). Please reload the code definition to get the latest changes.

Companies
Facebook   Amazon   LinkedIn   Snapchat   Yelp

Topics

291. Word Pattern II (Hard)

Given a `pattern` and a string `str`, find if `str` follows the same pattern.

Here follow means a full match, such that there is a bijection between a letter in `pattern` and a non-empty substring in `str`.

Examples:

1. pattern = `"abab"`, str = `"redblueredblue"` should return true.
2. pattern = `"aaaa"`, str = `"asdasdasdasd"` should return true.
3. pattern = `"aabb"`, str = `"xyzabcxzyabc"` should return false.

Notes:
You may assume both `pattern` and `str` contains only lowercase letters.

Companies

Uber   Dropbox

Topics

Backtracking

290. Word Pattern (Easy)

Given a `pattern` and a string `str`, find if `str` follows the same pattern.

Here follow means a full match, such that there is a bijection between a letter in `pattern` and a non-empty word in `str`.

Examples:

1. pattern = `"abba"`, str = `"dog cat cat dog"` should return true.
2. pattern = `"abba"`, str = `"dog cat cat fish"` should return false.
3. pattern = `"aaaa"`, str = `"dog cat cat dog"` should return false.
4. pattern = `"abba"`, str = `"dog dog dog dog"` should return false.

Notes:
You may assume `pattern` contains only lowercase letters, and `str` contains lowercase letters separated by a single space.

Companies

Uber Dropbox

Topics

Hash Table

212. Word Search II (Hard)

Given a 2D board and a list of words from the dictionary, find all words in the board.

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

For example,
Given words = `["oath","pea","eat","rain"]` and board =

```
[
  ['o','a','a','n'],
  ['e','t','a','e'],
  ['i','h','k','r'],
  ['i','f','l','v']
]
```

Return `["eat","oath"]`.

Note:
You may assume that all inputs are consist of lowercase letters `a-z`.

You would need to optimize your backtracking to pass the larger test. Could you stop backtracking earlier?

If the current candidate does not exist in all words' prefix, you could stop backtracking immediately. What kind of data structure could answer such query efficiently? Does a hash table work? Why or why not? How about a Trie? If you would like to learn how to implement a basic trie, please work on this problem: Implement Trie (Prefix Tree) (https://leetcode.com/problems/implement-trie-prefix-tree/) first.

Companies

Google   Microsoft   Airbnb

Topics

Backtracking   Trie

79. Word Search (Medium)

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 cell, where "adjacent" cells are those horizontally or vertically neighboring. The same letter cell may not be used more than once.

For example,
Given board =

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

word = "ABCCED", -> returns `true`,

word = "SEE", -> returns `true`,

word = "ABCB", -> returns `false`.

Companies

Facebook   Microsoft   Bloomberg

Topics

Array   Backtracking

6. ZigZag Conversion (Medium)

The string `"PAYPALISHIRING"` is written in a zigzag pattern on a given number of rows like this: (you may want to display this pattern in a fixed font for better legibility)

```
P   A   H   N
A P L S I I G
Y   I   R
```

And then read line by line: `"PAHNAPLSIIGYIR"`

Write the code that will take a string and make this conversion given a number of rows:

```
string convert(string text, int nRows);
```

`convert("PAYPALISHIRING", 3)` should return `"PAHNAPLSIIGYIR"`.

Companies

Topics

String

281. Zigzag Iterator (Medium)

Given two 1d vectors, implement an iterator to return their elements alternately.

For example, given two 1d vectors:

```
v1 = [1, 2]
v2 = [3, 4, 5, 6]
```

By calling next repeatedly until hasNext returns `false`, the order of elements returned by next should be: `[1, 3, 2, 4, 5, 6]`.

Follow up : What if you are given `k` 1d vectors? How well can your code be extended to such cases?

Clarification for the follow up question -   Update (2015-09-18):
The "Zigzag" order is not clearly defined and is ambiguous for `k > 2` cases. If "Zigzag" does not look right to you, replace "Zigzag" with "Cyclic". For example, given the following input:

```
[1,2,3]
[4,5,6,7]
[8,9]
```

It should return `[1,4,8,2,5,9,3,6,7]`.

Companies

Google

Topics

Design