|  |  |
| --- | --- |
| GITAM LOGO.png | **GCGC-Competence Development Assessments on CT3 Platform** |

**Question – 1**

**(Gas Station)**

**Problem Statement:**

Given two integer arrays A and B of size N. There are N gas stations along a circular route, where the amount of gas at station i is A[i].

You have a car with an unlimited gas tank and it costs B[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 minimum starting gas station's index if you can travel around the circuit once, otherwise return -1.

You can only travel in one direction. i to i+1, i+2, ... n-1, 0, 1, 2.. Completing the circuit means starting at i and ending up at i again.

Input Format

The first argument given is the integer array A. The second argument given is the integer array B.

Output Format

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

Example Input

A = [1, 2] B = [2, 1]

Example Output

1

**Question – 2**

**(Majority Element)**

**Problem Statement:**

Given an array of size n, find the majority element. The majority element is the element that appears more than floor(n/2) times.

You may assume that the array is non-empty, and the majority element always exist in the array.

Example:

Input : [2, 1, 2]

Return : 2 which occurs 2 times which is greater than 3/2.

**Question – 3**

**Distribute Candy**

**Problem Statement:**

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:

1. Each child must have at least one candy.

2. Children with a higher rating get more candies than their neighbours.

What is the minimum candies you must give?

**Input Format:**

The first and the only argument contains N integers in an array A.

**Output Format:**

Return an integer, representing the minimum candies to be given.

**Example:**

Input 1:

A = [1, 2]

Output 1:

3

**Question – 4**

**Problem Statement:**

Find the **longest increasing subsequence** of a given array of integers, **A**.

In other words, find a subsequence of array in which the subsequence’s elements are in strictly increasing order, and in which the subsequence is as long as possible.   
This subsequence is not necessarily contiguous, or unique.  
In this case, we only care about the **length** of the longest increasing subsequence.

**Input Format:**

The first and the only argument is an integer array A.

**Output Format:**

Return an integer representing the length of the longest increasing subsequence.

**Constraints:**

1 <= length(A) <= 2500

1 <= A[i] <= 2000

**Example :**

Input 1:

A = [1, 2, 1, 5]

Output 1:

3

**Question – 5**

**Unique Binary Search Trees**

**Problem Statement:**

Given **A**, generate all structurally unique BST’s (binary search trees) that store values **1…A**.

**Input Format:**

The first and the only argument of input contains the integer, A.

**Output Format:**

Return a list of tree nodes representing the generated BST's.

**Constraints:**

1 <= A <= 15

**Example:**

Input 1:

A = 3

Output 1:

**Question – 6**

**Max Rectangle in Binary Matrix**

**Problem Statement:**

Given a 2D binary matrix filled with 0’s and 1’s, find the largest rectangle containing **all ones** and return its area.

Bonus if you can solve it in O(n^2) or less.

**Example :**

A : [ 1 1 1

0 1 1

1 0 0

]

Output : 4

**Question – 7**

# Distinct Subsequences

**Problem Statement:**

Given two sequences **A**, **B**, count number of unique ways in sequence **A**, to form a subsequence that is identical to the sequence **B**.

**Subsequence :** 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).

**Input Format:**

The first argument of input contains a string, A.

The second argument of input contains a string, B.

**Output Format:**

Return an integer representing the answer as described in the problem statement.

**Constraints:**

1 <= length(A), length(B) <= 700

**Example :**

Input 1:

A = "abc"

B = "abc"

Output 1:

1

**Question – 8**

# Unique Paths in a Grid

**Problem Statement:**

Given a grid of size m \* n, lets assume you are starting at (1,1) and your goal is to reach (m,n). At any instance, if you are on (x,y), you can either go to (x, y + 1) or (x + 1, y).

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.

**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.

**Question – 9**

# Max Product Subarray

**Problem Statement:**

Find the contiguous subarray within an array (containing at least one number) which has the largest product.  
Return an integer corresponding to the maximum product possible.

**Example :**

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

Return : 6

Possible with [2, 3]

**Question – 10**

# Ways to Decode

**Problem Statement:**

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 **A** containing digits, determine the total number of ways to decode it modulo **109 + 7**.

**Problem Constraints**

1 <= |A| <= 105

**Input Format**

The first and the only argument is a string **A**.

**Output Format**

Return a single integer denoting the total number of ways to decode it modulo **109 + 7**.

**Example Input**

Input 1:

A = "8"

Input 2:

A = "12"

**Example Output**

Output 1:

1

Output 2:

2

**Question – 11**

# Best Time to Buy and Sell Stocks I

**Problem Description**

Say you have an array, **A**, for which the **i**th element is the price of a given stock on day **i**.

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

Return the **maximum** possible profit.

**Problem Constraints**

0 <= len(A) <= 7e5

1 <= A[i] <= 1e7

**Input Format**

The first and the only argument is an array of integers, A.

**Output Format**

Return an integer, representing the maximum possible profit.

**Example Input**

Input 1:

A = [1, 2]

Input 2:

A = [1, 4, 5, 2, 4]

**Example Output**

Output 1:

1

Output 2:

4

**Question – 12**

# Best Time to Buy and Sell Stocks II

**Problem Description**

Say you have an array, **A**, 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 (i.e., 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).

**Input Format:** The first and the only argument is an array of integer, A.

**Output Format:** Return an integer, representing the maximum possible profit.

**Constraints:** 0 <= len(A) <= 1e5 1 <= A[i] <= 1e7 **Example:**

Input 1:

A = [1, 2, 3]

Output 1:

2

**Question – 13**

# Best Time to Buy and Sell Stocks III

Say you have an array, **A**, 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 **2** transactions.

Return the maximum possible profit.

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

**Input Format:**

The first and the only argument is an integer array, A.

**Output Format:**

Return an integer, representing the maximum possible profit.

**Constraints:**

1 <= length(A) <= 7e5

1 <= A[i] <= 1e7

**Examples:**

Input 1:

A = [1, 2, 1, 2]

Output 1:

2

Explanation 1:

Day 0 : Buy

Day 1 : Sell

Day 2 : Buy

Day 3 : Sell

Input 2:

A = [7, 2, 4, 8, 7]

Output 2:

6

**Question – 14**

# Max Sum Path in Binary Tree

Given a binary tree **T**, find the maximum path sum.

The path may start and end at any node in the tree.

**Input Format:**

The first and the only argument contains a pointer to the root of T, A.

**Output Format:**

Return an integer representing the maximum sum path.

**Constraints:**

1 <= Number of Nodes <= 7e4

-1000 <= Value of Node in T <= 1000

**Question – 15**

# Regular Expression Match

Implement wildcard pattern matching with support for ‘?’ and ‘\*’ for strings **A** and **B**.

* ’?’ : Matches any single character.
* ‘\*’ : Matches any sequence of characters (including the empty sequence).

The matching should cover the entire input string (not partial).

**Input Format:**

The first argument of input contains a string A.

The second argument of input contains a string B.

**Output Format:**

Return 0 or 1:

=> 0 : If the patterns do not match.

=> 1 : If the patterns match.

**Constraints:**

1 <= length(A), length(B) <= 9e4

**Examples :**

Input 1:

A = "aa"

B = "a"

Output 1:

0

Input 2:

A = "aa"

B = "aa"

Output 2:

1

Input 3:

A = "aaa"

B = "aa"

Output 3:

0

Input 4:

A = "aa"

B = "\*"

Output 4:

1

Input 5:

A = "aa"

B = "a\*"

Output 5:

1

Input 6:

A = "ab"

B = "?\*"

Output 6:

1

Input 7:

A = "aab"

B = "c\*a\*b"

Output 7:

0

**Question – 16**

# Palindrome Partitioning II

Given a string **A**, partition **A** such that every substring of the partition is a palindrome.

Return the **minimum** cuts needed for a palindrome partitioning of **A**.

**Input Format:**

The first and the only argument contains the string A.

**Output Format:**

Return an integer, representing the answer as described in the problem statement.

**Constraints:**

1 <= length(A) <= 501

**Examples:**

Input 1:

A = "aba"

Output 1:

0

**Question – 17**

# Min Sum Path in Matrix

**Problem Description**

Given a 2D integer array **A** of size M x N, you need to 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.

**Input Format**

First and only argument is an 2D integer array **A** of size M x N.

**Output Format**

Return a single integer denoting the minimum sum of a path from cell (1, 1) to cell (M, N).

**Example Input**

Input 1:

A = [ [1, 3, 2]

[4, 3, 1]

[5, 6, 1]

]

**Example Output**

Output 1:

9

**Question – 18**

# Min Jumps Array

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

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

Return the **minimum** number of jumps required to reach the last index.

If it is not possible to reach the last index, return -1.

**Input Format:**

The first and the only argument contains an integer array, A.

**Output Format:**

Return an integer, representing the answer as described in the problem statement.

**Constraints:**

1 <= N <= 1e6

0 <= A[i] <= 50000

**Examples:**

Input 1:

A = [2, 1, 1]

Output 1:

1

Explanation 1:

The shortest way to reach index 2 is

Index 0 -> Index 2

that requires only 1 jump.

Input 2:

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

Output 2:

2

**Question – 19**

# Edit Distance

Given two strings **A** and **B**, find the minimum number of steps required to convert **A** to **B**. (each operation is counted as 1 step.)

You have the following 3 operations permitted on a word:

* Insert a character
* Delete a character
* Replace a character

**Input Format:**

The first argument of input contains a string, A.

The second argument of input contains a string, B.

**Output Format:**

Return an integer, representing the minimum number of steps required.

**Constraints:**

1 <= length(A), length(B) <= 450

**Examples:**

Input 1:

A = "abad"

B = "abac"

Output 1:

1

**Question – 20**

# Unique Binary Search Trees II

Given an integer **A**, how many structurally unique BST’s (binary search trees) exist that can store values **1…A**?

**Input Format:**

The first and the only argument of input contains the integer, A.

**Output Format:**

Return an integer, representing the answer asked in problem statement.

**Constraints:**

1 <= A <= 18

**Example:**

Input 1:

A = 3

Output 1:

5

**Question – 21**

# Word Break

Given a string **A** and a dictionary of words **B**, determine if **A** can be segmented into a space-separated sequence of one or more dictionary words.

**Input Format:**

The first argument is a string, A.

The second argument is an array of strings, B.

**Output Format:**

Return 0 / 1 ( 0 for false, 1 for true ) for this problem.

**Constraints:**

1 <= len(A) <= 6500

1 <= len(B) <= 10000

1 <= len(B[i]) <= 20

**Examples:**

Input 1:

A = "myinterviewtrainer",

B = ["trainer", "my", "interview"]

Output 1:

1

Explanation 1:

Return 1 ( corresponding to true ) because "myinterviewtrainer" can be segmented as "my interview trainer".

Input 2:

A = "a"

B = ["aaa"]

Output 2:

0

**Question – 22**

# Regular Expression II

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:

int isMatch(const char \*s, const char \*p)

Some examples:

isMatch("aa","a") → 0

isMatch("aa","aa") → 1

isMatch("aaa","aa") → 0

isMatch("aa", "a\*") → 1

isMatch("aa", ".\*") → 1

isMatch("ab", ".\*") → 1

isMatch("aab", "c\*a\*b") → 1

Return 0 / 1 ( 0 for false, 1 for true ) for this problem

**Question – 23**

# Jump Game Array

Given an array of non-negative integers, **A**, you are initially positioned at the 0th 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.

**Input Format:**

The first and the only argument of input will be an integer array A.

**Output Format:**

Return an integer, representing the answer as described in the problem statement.

=> 0 : If you cannot reach the last index.

=> 1 : If you can reach the last index.

**Constraints:**  
1 <= len(A) <= 106  
  
0 <= A[i] <= 30

**Examples:**

Input 1:

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

Output 1:

1

**Question – 24**

# Interleaving Strings

Given **A**, **B**, **C**, find whether **C** is formed by the interleaving of **A** and **B**.

**Input Format:**\*

The first argument of input contains a string, A.

The second argument of input contains a string, B.

The third argument of input contains a string, C.

**Output Format:**

Return an integer, 0 or 1:

=> 0 : False

=> 1 : True

**Constraints:**

1 <= length(A), length(B), length(C) <= 150

**Examples:**

Input 1:

A = "aabcc"

B = "dbbca"

C = "aadbbcbcac"

Output 1:

1

Explanation 1:

"aa" (from A) + "dbbc" (from B) + "bc" (from A) + "a" (from B) + "c" (from A)

Input 2:

A = "aabcc"

B = "dbbca"

C = "aadbbbaccc"

Output 2:

0

**Question – 25**

# Word Break II

Given a string **A** and a dictionary of words **B**, add spaces in **A** to construct a sentence where each word is a valid dictionary word.

Return all such possible sentences.

**Note :** Make sure the strings are sorted in your result.

**Input Format:**

The first argument is a string, A.

The second argument is an array of strings, B.

**Output Format:**

Return a vector of strings representing the answer as described in the problem statement.

**Constraints:**

1 <= len(A) <= 50

1 <= len(B) <= 25

1 <= len(B[i]) <= 20

**Examples:**

Input 1:

A = "b"

B = ["aabbb"]

Output 1:

[]

Input 1:

A = "catsanddog",

B = ["cat", "cats", "and", "sand", "dog"]

Output 1:

["cat sand dog", "cats and dog"]

**Question – 26**

# Longest valid Parentheses

Given a string **A** containing just the characters **’(‘** and **’)’**.

Find the length of the **longest** valid (well-formed) parentheses substring.

**Input Format:**

The only argument given is string A.

**Output Format:**

Return the length of the longest valid (well-formed) parentheses substring.

**Constraints:**

1 <= length(A) <= 750000

**For Example**

Input 1:

A = "(()"

Output 1:

2

**Question – 27**

# Stairs

You are climbing a stair case and it takes **A** 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?

**Input Format:**

The first and the only argument contains an integer A, the number of steps.

**Output Format:**

Return an integer, representing the number of ways to reach the top.

**Constrains:**

1 <= A <= 36

**Example :**

Input 1:

A = 2 Output 1:

2 Explanation 1:

[1, 1], [2] Input 2:

A = 3 Output 2:

3 Explanation 2:

[1 1 1], [1 2], [2 1]

**Question – 28**

# Swap List Nodes in pairs

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.

**Question – 29**

# Rotate List

Given a list, rotate the list to the right by k places, where k is non-negative.

For example:

Given 1->2->3->4->5->NULL and k = 2,  
return 4->5->1->2->3->NULL.

**Question – 30**

# Reorder List

Given a singly linked list

L: L0 → L1 → … → Ln-1 → Ln,

reorder it to:

L0 → Ln → L1 → Ln-1 → L2 → Ln-2 → …

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}.

**Question – 31**

# Sort List

Sort a linked list in O(n log n) time using constant space complexity.

**Example :**

Input : 1 -> 5 -> 4 -> 3

Returned list : 1 -> 3 -> 4 -> 5

**Question – 32**

# Remove Duplicates from Sorted List II

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.

**Question – 33**

# Merge Two Sorted Lists

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, and should also be sorted.

For example, given following linked lists :

5 -> 8 -> 20

4 -> 11 -> 15

The merged list should be :

4 -> 5 -> 8 -> 11 -> 15 -> 20

**Question – 34**

# Remove Duplicates from Sorted List

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.

**Question – 35**

# Add Two Numbers as Lists

You are given two linked lists representing two non-negative numbers. 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.

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

342 + 465 = 807

Make sure there are no trailing zeros in the output list  
So, 7 -> 0 -> 8 -> 0 is not a valid response even though the value is still 807.

**Question – 36**

# Remove Nth Node from List End

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 becomes 1->2->3->5.

**Note:**

If n is greater than the size of the list, remove the first node of the list.

Try doing it using constant additional space.

**Question – 37**

# Partition List

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.

**Question – 38**

# Insertion Sort List

Sort a linked list using insertion sort.

We have explained Insertion Sort at Slide 7 of [Arrays](http://www.interviewbit.com/courses/programming/topics/arrays/)

[Insertion Sort Wiki](http://en.wikipedia.org/wiki/Insertion_sort#Algorithm) has some details on Insertion Sort as well.

**Example :**

Input : 1 -> 3 -> 2

Return 1 -> 2 -> 3

**Question – 39**

# List Cycle

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

Try solving it using constant additional space.

**Example:**

Input:

\_\_\_\_\_\_

| |

\/ |

1 -> 2 -> 3 -> 4

Return the node corresponding to node 3.

**Question – 40**

# Intersection of Linked Lists

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.

**Question – 41**

# Reverse Link List II

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.

**Note 2:**  
Usually the version often seen in the interviews is reversing the whole linked list which is obviously an easier version of this question.

**Question – 42**

# Evaluate Expression

**Problem Description**

An arithmetic expression is given by a charater array A of size N. Evaluate the value of an arithmetic expression in Reverse Polish Notation.

Valid operators are +, -, \*, /. Each character may be an integer or an operator.

**Problem Constraints**

1 <= N <= 105

**Input Format**

The only argument given is character array A.

**Output Format**

Return the value of arithmetic expression formed using reverse Polish Notation.

**Example Input**

Input 1:

A = ["2", "1", "+", "3", "*"]*

*Input 2:*

*A = ["4", "13", "5", "/", "+"]*

*Example Output*

*Output 1:*

*9*

*Output 2:*

*6*

**Question – 43**

# Rain Water Trapped

**Problem Description**

Given an integer array **A** of 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.

**Problem Constraints**

1 <= **|A|** <= 100000

**Input Format**

The only argument given is integer array A.

**Output Format**

Return the total water it is able to trap after raining.

**Example Input**

Input 1:

A = [0, 1, 0, 2, 1, 0, 1, 3, 2, 1, 2, 1]

Input 2:

A = [1, 2]

**Example Output**

Output 1:

6

Output 2:

0

**Question – 44**

# Generate all Parentheses

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.

Return 0 / 1 ( 0 for false, 1 for true ) for this problem

PROBLEM APPROACH :

Complete solution in hints.

**Question – 45**

# Largest Rectangle in Histogram

**Problem Description**

Given an array of integers **A** .

A represents a histogram i.e A[i] denotes height of the ith histogram's bar. Width of each bar is 1.

Find the area of the largest rectangle formed by the histogram.

**Problem Constraints**

1 <= |A| <= 100000

1 <= A[i] <= 1000000000

**Input Format**

The only argument given is the integer array A.

**Output Format**

Return the area of largest rectangle in the histogram.

**Example Input**

Input 1:

A = [2, 1, 5, 6, 2, 3]

Input 2:

A = [2]

**Example Output**

Output 1:

10

Output 2:

2

**Question – 46**

# Sliding Window Maximum

Given an array of integers **A**. There is a sliding window of size **B** which   
is moving from the very left of the array to the very right.   
You can only see the w numbers in the window. Each time the sliding window moves   
rightwards by one position. You have to find the maximum for each window.   
The following example will give you more clarity.

The array **A** is [1 3 -1 -3 5 3 6 7], and **B** is 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 |

Return an array **C**, where **C[i]** is the maximum value of from **A[i]** to **A[i+B-1]**.

**Note**: If **B** > length of the array, return 1 element with the max of the array.

**Input Format**

The first argument given is the integer array A.

The second argument given is the integer B.

**Output Format**

Return an array C, where C[i] is the maximum value of from A[i] to A[i+B-1]

**For Example**

Input 1:

A = [1, 3, -1, -3, 5, 3, 6, 7]

B = 3

Output 1:

C = [3, 3, 5, 5, 6, 7]

**Question – 47**

# Simplify Directory Path

Given a string **A** representing an absolute path for a file (Unix-style).

Return the string A after simplifying the absolute path.

**Note**:

1. Absolute path always begin with **’/’** ( root directory ).
2. Path will not have whitespace characters.

**Input Format**

The only argument given is string A.

**Output Format**

Return a string denoting the simplified absolue path for a file (Unix-style).

**For Example**

Input 1:

A = "/home/"

Output 1:

"/home"

Input 2:

A = "/a/./b/../../c/"

Output 2:

"/c"

**Question – 48**

# Min Stack

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.

Note that all the operations have to be constant time operations.

Questions to ask the interviewer :

Q: What should getMin() do on empty stack?

A: In this case, return -1.

Q: What should pop do on empty stack?

A: In this case, nothing.

Q: What should top() do on empty stack?

A: In this case, return -1

**NOTE** : If you are using your own declared global variables, make sure to clear them out in the constructor.

**Question – 49**

# Kth Row of Pascal's Triangle

**Problem Description**

Given an index k, return the kth row of the Pascal's triangle.

Pascal's triangle: To generate A[C] in row R, sum up A'[C] and A'[C-1] from previous row R - 1.

**Example:**

Input : k = 3

Return : [1,3,3,1]

**Note**: k is 0 based. k = 0, corresponds to the row [1].

**Note**: Could you optimize your algorithm to use only O(k) extra space?

**Question – 50**

# Rotate Matrix

You are given an n x n 2D matrix representing an image.

Rotate the image by 90 degrees (clockwise).

You need to do this in place.

Note that if you end up using an additional array, you will only receive partial score.

**Example:**

If the array is

[

[1, 2],

[3, 4]

]

Then the rotated array becomes:

[

[3, 1],

[4, 2]

]

**Question – 51**

# Max Sum Contiguous Subarray

Find the **contiguous** subarray within an array, **A** of length **N** which has the **largest sum**.

**Input Format:**

The first and the only argument contains an integer array, A.

**Output Format:**

Return an integer representing the maximum possible sum of the contiguous subarray.

**Constraints:**

1 <= N <= 1e6

-1000 <= A[i] <= 1000

**For example:**

Input 1:

A = [1, 2, 3, 4, -10]

Output 1:

10

Explanation 1:

The subarray [1, 2, 3, 4] has the maximum possible sum of 10.

Input 2:

A = [-2, 1, -3, 4, -1, 2, 1, -5, 4]

Output 2:

6

**Question – 52**

# Find Duplicate in Array

**Problem Description**

Given a read only array of n + 1 integers between 1 and n, find one number that repeats in linear time using less than O(n) space and traversing the stream sequentially O(1) times.

**Sample Input:** [3 4 1 4 1] **Sample Output:** 1 If there are multiple possible answers ( like in the sample case above ), output any one.

If there is no duplicate, output -1

**Question – 53**

# Merge Intervals

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] would result in [1,5],[6,9].

**Example 2:**

Given [1,2],[3,5],[6,7],[8,10],[12,16], insert and merge [4,9] would result in [1,2],[3,10],[12,16].

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

**Question – 54**

# Spiral Order Matrix I

Given a matrix of m \* n elements (m rows, n columns), return all elements of the matrix in spiral order.

**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]

**Question – 55**

# Repeat and Missing Number Array

There are certain problems which are asked in the interview to also check how you take care of **overflows** in your problem.  
This is one of those problems.  
Please take extra care to make sure that you are type-casting your ints to long properly and at all places. **Try to verify if your solution works if number of elements is as large as 105**

**Food for thought** :

* Even though it might not be required in this problem, in some cases, you might be required to order the operations cleverly so that the numbers do not overflow.  
  For example, if you need to calculate n! / k! where n! is factorial(n), one approach is to calculate factorial(n), factorial(k) and then divide them.  
  Another approach is to only multiple numbers from k + 1 ... n to calculate the result.  
  Obviously approach 1 is more susceptible to overflows.

You are given a read only array of n integers from 1 to n.

Each integer appears exactly once except A which appears twice and B which is missing.

Return A and B.

*Note: Your algorithm should have a linear runtime complexity. Could you implement it without using extra memory?*

*Note that in your output A should precede B.*

**Example:**

Input:[3 1 2 5 3]

Output:[3, 4]

A = 3, B = 4

**Question – 56**

# Merge Overlapping Intervals

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].

Make sure the returned intervals are sorted

**Question – 57**

# Set Matrix Zeros

**Problem Description**

Given a matrix, **A** of size **M** x **N** of **0s** and **1s**. If an element is **0**, set its entire row and column to **0**.

**Note**: This will be evaluated on the extra memory used. Try to minimize the space and time complexity.

**Input Format:**

The first and the only argument of input contains a 2-d integer matrix, A, of size M x N.

**Output Format:**

Return a 2-d matrix that satisfies the given conditions.

**Constraints:**

1 <= N, M <= 1000

0 <= A[i][j] <= 1

**Examples:**

Input 1:

[ [1, 0, 1],

[1, 1, 1],

[1, 1, 1] ]

Output 1:

[ [0, 0, 0],

[1, 0, 1],

[1, 0, 1] ]

Input 2:

[ [1, 0, 1],

[1, 1, 1],

[1, 0, 1] ]

Output 2:

[ [0, 0, 0],

[1, 0, 1],

[0, 0, 0] ]

**Question – 58**

# Spiral Order Matrix II

Given an integer **A**, generate a square matrix filled with elements from **1** to **A2** in **spiral order**.

**Input Format:**

The first and the only argument contains an integer, A.

**Output Format:**

Return a 2-d matrix of size A x A satisfying the spiral order.

**Constraints:**

1 <= A <= 1000

**Examples:**

Input 1:

A = 3

Output 1:

[ [ 1, 2, 3 ],

[ 8, 9, 4 ],

[ 7, 6, 5 ] ]

Input 2:

4

Output 2:

[ [1, 2, 3, 4],

[12, 13, 14, 5],

[11, 16, 15, 6],

[10, 9, 8, 7] ]

**Question – 59**

# Largest Number

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.

**Question – 60**

# First Missing Integer

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

**Example:**

Given [1,2,0] return 3,

[3,4,-1,1] return 2,

[-8, -7, -6] returns 1

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

**Question – 61**

# Add One To Number

**Problem Description**

Given a **non-negative** number represented as an array of digits, add **1** to the number ( increment the number represented by the digits ).

The digits are stored such that the most significant digit is at the head of the list.

**NOTE:** Certain things are intentionally left unclear in this question which you should practice asking the interviewer. For example: for this problem, following are some good questions to ask :

* **Q :** Can the input have **0's** before the most significant digit. Or in other words, is **0 1 2 3** a valid input?
* **A :** For the purpose of this question, **YES**
* **Q :** Can the output have **0's** before the most significant digit? Or in other words, is **0 1 2 4** a valid output?
* **A :** For the purpose of this question, **NO**. Even if the input has zeroes before the most significant digit.

**Input Format**

First argument is an array of digits.

**Output Format**

Return the array of digits after adding one.

**Example Input**

Input 1:

[1, 2, 3]

**Example Output**

Output 1:

[1, 2, 4]

**Question – 62**

# N/3 Repeat Number

**Problem Description**

You're given a read only array of n integers. Find out if any integer occurs more than n/3 times in the array in linear time and constant additional space.

If so, return the integer. If not, return -1.

If there are multiple solutions, return any one.

**Example:**

Input: [1 2 3 1 1]

Output: 1

1 occurs 3 times which is more than 5/3 times.

**Question – 63**

# Pascal Triangle

**Problem Description**

Given numRows, generate the first numRows of Pascal's triangle.

Pascal's triangle : To generate A[C] in row R, sum up A'[C] and A'[C-1] from previous row R - 1.

**Example:**

Given numRows = 5,

Return

[

[1],

[1,1],

[1,2,1],

[1,3,3,1],

[1,4,6,4,1]

]

**Constraints:**  
0 <= numRows <= 25

**Question – 64**

# Maximum Consecutive Gap

Given an unsorted array, find the maximum difference between the successive elements in its sorted form.

Try to solve it in linear time/space.

**Example :**

Input : [1, 10, 5]

Output : 5

**Return 0 if the array contains less than 2 elements.**

* You may assume that all the elements in the array are non-negative integers and fit in the 32-bit signed integer range.
* You may also assume that the difference will not overflow.

**Question – 65**

# Max Distance

**Problem Description**

Given an array **A** of integers, find the maximum of **j - i** subjected to the constraint of **A[i] <= A[j]**.

**Input Format**

First and only argument is an integer array A.

**Output Format**

Return an integer denoting the maximum value of j - i;

**Example Input**

Input 1:

A = [3, 5, 4, 2]

**Example Output**

Output 1:

2

**Example Explanation**

Explanation 1:

Maximum value occurs for pair (3, 4).

**Question – 66**

# Next Permutation

mplement the **next permutation**, which rearranges numbers into the numerically next greater permutation of numbers for a given array **A** of size **N**.

If such arrangement is not possible, it must be rearranged as the lowest possible order *i.e., sorted in an ascending order*.

**Note:**

1. The replacement must be in-place, do \*\*not\*\* allocate extra memory.

2. DO NOT USE LIBRARY FUNCTION FOR NEXT PERMUTATION. Use of Library functions will disqualify your submission retroactively and will give you penalty points.

**Input Format:**

The first and the only argument of input has an array of integers, A.

**Output Format:**

Return an array of integers, representing the next permutation of the given array.

**Constraints:**

1 <= N <= 5e5

1 <= A[i] <= 1e9

**Examples:**

Input 1:

A = [1, 2, 3]

Output 1:

[1, 3, 2]

Input 2:

A = [3, 2, 1]

Output 2:

[1, 2, 3]

Input 3:

A = [1, 1, 5]

Output 3:

[1, 5, 1]

Input 4:

A = [20, 50, 113]

Output 4:

[20, 113, 50]

**Question – 67**

# Palindrome Integer

**Problem Description**

Determine whether an integer is a palindrome. Do this without extra space.

A palindrome integer is an integer x for which reverse(x) = x where reverse(x) is x with its digit reversed. Negative numbers are not palindromic.

**Example** :

Input : 12121

Output : 1

Input : 123

Output : 0

**Question – 68**

# Verify Prime

Given a number N, verify if N is prime or not.

Return 1 if N is prime, else return 0.

**Example :**

Input : 7

Output : True

**Question – 69**

# Excel Column Number

**Problem Description**

Given a column title **A** as appears in an Excel sheet, return its corresponding column number.

**Problem Constraints**

1 <= |A| <= 100

**Input Format**

First and only argument is string A.

**Output Format**

Return an integer

**Example Input**

Input 1:

"A"

Input 2:

"AB"

**Example Output**

Output 1:

1

Output 2:

28

**Question – 70**

# Reverse integer

**Problem Description**

You are given an integer **N** and the task is to reverse the digits of the given integer. Return **0** if the result overflows and does not fit in a 32 bit signed integer  
  
Look at the example for clarification.

**Problem Constraints**

N belongs to the Integer limits.

**Input Format**

Input an Integer.

**Output Format**

Return a single integer denoting the reverse of the given integer.

**Example Input**

Input 1:

x = 123

Input 2:

x = -123

**Example Output**

Output 1:

321

Ouput 2:

-321

**Question – 71**

# Excel Column Title

**roblem Description**

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

**Problem Constraints**

1 <= A <= 1000000000

**Input Format**

First and only argument is integer A.

**Output Format**

Return a string, the answer to the problem.

**Example Input**

Input 1:

A = 1

Input 2:

A = 28

**Example Output**

Output 1:

"A"

Output 2:

"AB"

**Question – 72**

# Grid Unique Paths

A robot is located at the top-left corner of an **A x B 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?

Note: A and B will be such that the resulting answer fits in a 32 bit signed integer.

**Example :**

Input : A = 2, B = 2

Output : 2

2 possible routes : (0, 0) -> (0, 1) -> (1, 1)

OR : (0, 0) -> (1, 0) -> (1, 1)

**Question – 73**

# Trailing Zeros in Factorial

**Problem Description**

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

**Note**: Your solution should be in logarithmic time complexity.

\*\*Problem Constraints\*\*

1 <= A <= 10000

\*\*Input Format\*\*

First and only argumment is integer A.

\*\*Output Format\*\*

Return an integer, the answer to the problem.

\*\*Example Input\*\*

Input 1:

A = 4

Input 2:

A = 5

\*\*Example Output\*\*

Output 1:

0

Output 2:

1

**Question – 74**

# All Factors

Given a number N, find all factors of N.

**Example:**

N = 6

factors = {1, 2, 3, 6}

Make sure the returned array is sorted.

**Problem Approach:**

VIDEO : https://www.youtube.com/watch?v=dolcMgiJ7I0

Complete code in the hint.

**Question – 75**

# Monkeys and Doors

There are 100 doors, all closed.   
In a nearby cage are 100 monkeys.

The first monkey is let out and runs along the doors opening every one.   
The second monkey is then let out and runs along the doors closing the 2nd, 4th, 6th,… - **all the even-numbered doors**.   
The third monkey is let out. He attends only to the 3rd, 6th, 9th,… doors **(every third door, in other words)**, closing any that is open and opening any that is closed, and so on.   
After all 100 monkeys have done their work in this way, what state are the doors in after the last pass?

**Answer with the number of open doors.**

*Answer is an integer.* Just put the number without any decimal places if it’s an integer. If the answer is Infinity, output ***Infinity***.

**Question – 76**

# Daughters' Ages

Two MIT math grads bump into each other at Fairway on the upper west side. They haven’t seen each other in over 20 years.

The first grad says to the second: “how have you been?”

Second: “great! i got married and i have three daughters now”

First: “really? how old are they?”

Second: “well, the product of their ages is 72, and the sum of their ages is the same as the number on that building over there..”

First: “right, ok.. oh wait.. hmm, i still don’t know”

Second: “oh sorry, the oldest one just started to play the piano”

First: “wonderful! my oldest is the same age!”

How old are the daughters?  
Respond with daughter ages sorted in ascending order and concatenated together. No spaces.

**Question – 77**

# Jelly Beans Jars

You have three jars that are all mislabeled. One contains peanut butter jelly beans, another grape jelly jelly beans and the third has a mix of both (not necessarily half-half mix).

How many minimum jelly beans would you have to pull out to find out how to fix the labels on the jars?

Labels on jars are as follows

Jar 1 : Peanut butter

Jar 2 : Grape

Jar 3 : P.b. / Grape

*Answer is a integer.* Just put the number without any decimal places if its an integer. If the answer is Infinity, output ***Infinity***.

**Question – 78**

# Cross the Bridge

Four people are on this side of the bridge. Everyone has to get across.   
Problem is that it’s dark and so you can’t cross the bridge without a flashlight and they only have one flashlight.   
Plus the bridge is only big enough for two people to cross at once.   
The four people walk at different speeds:

* one fella is so fast it only takes him 1 minute to cross the bridge,
* another 2 minutes,
* a third 5 minutes,
* the last it takes 10 minutes to cross the bridge.

When two people cross the bridge together (sharing the flashlight), they both walk at the slower person’s pace. What is the minimum time required for all 4 to cross the bridge.

Respond with a number which represents the number of minutes.

*Answer is a integer.* Just put the number without any decimal places if its an integer. If the answer is Infinity, output ***Infinity***.

**Question – 79**

# The Tribe

There are two tribes in Mars, Lie tribe and Truth Tribe.  
Lie tribe always speaks lie, True tribe always speaks truth.

You meet three mars people and ask from the first Person:

Q: What tribe do you belong to?

He replies something in his language which you don’t understand.

Second person tells you that he is saying that he belongs to Lie Tribe.  
Third person says that second person is lying.

What tribe does the third person belong to?

**Question – 80**

# Divide the Cake

Consider a rectangular cake with a rectangular section (of any size or orientation) removed from it. Is it possible to divide the cake exactly in half with only one cut?

**Question – 81**

Which offer is better?

1) You are to make a statement. If the statement is true, you get exactly $10. If the statement is false, you get either less than or more than $10 but not exactly $10.

2) You are to make a statement. Regardless of whether the statement is true or false, you get more than $10.

**Question – 82**

# Find the Defective Ball

You have 12 balls all look identical (in shape, color etc.).   
All of them have same weight except one defective ball.   
You don’t know that the defective one is heavier or lighter than other balls. You can use a two sided balance system (not the electronic one).

Find the minimum no. of measures required to separate the defective ball.

**Question – 83**

# Prisoners and Poison

A bad king has a cellar of 1000 bottles of delightful and very expensive wine.

A neighboring queen plots to kill the bad king and sends a servant to poison the wine.   
Fortunately (or say unfortunately) the bad king’s guards catch the servant after he has only poisoned one bottle.   
Alas, the guards don’t know which bottle but know that the poison is so strong that even if diluted 100,000 times it would still kill the king.

Furthermore, it takes one month to have an effect. The bad king decides he will get some of the prisoners in his vast dungeons to drink the wine. Being a clever bad king he knows he needs to murder as less prisoners as possible – believing he can fob off such a low death rate – and will still be able to drink the rest of the wine (999 bottles) at his anniversary party in 5 weeks time.

In the worst case, what is the minimum number of prisoner he would have to kill in order to find out the poisoned bottle? Do note that the king wants to minimize the number of prisoners involved in the experiment. He might decide to kill every prisoner involved in the experiment if he feels that they may tell the world about his evil plans.

**Question – 84**

# World Trips

Consider three identical airplanes starting at the same airport. Each plane has a fuel tank that holds just enough fuel to allow the plane to travel half the distance around the world. These airplanes possess the special ability to transfer fuel between their tanks in mid-flight.   
What are the maximum around the world trips that airplane1 can make?

* **Case 1 :** *Answer is a integer.* Just put the number without any decimal places if its an integer. If the answer is Infinity, output ***Infinity***.
* **Case 2 :** *Floating point number.* Round it off to 2 decimal places and output it as I.xx where I is the integer part of the answer, and xx are 2 decimal digits after rounding off.

**Question – 85**

# Color of the Bear

A Bear has fallen from a height of 10m from ground. It reached ground in sqrt(2) seconds. Luckily it didn’t get hurt. What color is the bear?

Just output the color. For example, following are acceptable answers.

* White
* Black
* Pink
* Brown
* Yellow
* Red
* Blue
* Green

**Question – 86**

# One Mile on the Globe

How many points are there on the globe where by walking one mile south, one mile east and one mile north you reach the place where you started?

* **Case 1 :** *Answer is a function of N.* Output it with N with multipliers or divisor numbers following N. For example, N \* 2, N / 2, N \* 3 / 4.
* **Case 2 :** *Answer is a integer.* Just put the number without any decimal places if its an integer. If the answer is Infinity, output ***Infinity***.
* **Case 3 :** *Floating point number.* Round it off to 2 decimal places and output it as I.xx where I is the integer part of the answer, and xx are 2 decimal digits after rounding off.

**Question – 87**

# Divide Gold Bar

You’ve got someone working for you for seven days and a gold bar to pay him. The gold bar is segmented into seven connected pieces.

Diagram

Description automatically generated

You must give them a piece of gold at the end of every day. What are the fewest number of cuts to the bar of gold that will allow you to pay him 1/7th each day?

*Answer is a integer.* Just put the number without any decimal places if its an integer. If the answer is Infinity, output ***Infinity***.

**Question – 88**

# Quarters on a Table

Consider a two-player game played on a circular table of unspecified diameter.

Shape, circle

Description automatically generated

Each player has an infinite supply of coins, and take turns placing a coin on the table such that it is completely on the table and does not overlap with any other coins already played.

**A player wins if he makes the last legal move.**

Which player (if any) has a strategy that will guarantee a win? Assume that the diameter of the table is greater than the diameter of the coin.

**Question – 89**

# Measure Milk by Cans

There is a drum full of milk.   
People come for buying milk in the range of 1-40 litres.   
You can have only 4 cans to draw milk out of drum. Tell me what should be the measurement of these four cans so that you can measure any amount of milk in the range of 1-40 litres.

Note that the cans cannot be used more than once to refill from the drum.

**Output the numbers in ascending order space separated.**

**Question – 90**

# Light Switches in the Cellar

In your cellar there are three light switches in the OFF position. Each switch controls one of three light bulbs on floor above.   
You may move any of the switches but you may only go upstairs to inspect the bulbs.   
When upstairs, you cannot access the switches. What is the minimum number of times you need to go upstairs to determine the switch for each bulb?

Since your answer is a integer, just put the number without any decimal places if its an integer. If the answer is *Infinity*, output **Infinity.**

**Question – 91**

# Next Number II

Identify the next number in the sequence

31, 28, 31, 30, \_\_?

*Answer is a integer.* Just put the number without any decimal places if its an integer. If the answer is Infinity, output ***Infinity***.

**Question – 92**

# Eggs and Building

There is a building of 100 floors

If an egg drops from the Nth floor or above it will break. If it’s dropped from any floor below, it will not break. You’re given 2 eggs. Find N, while minimizing the number of drops for the worst case.

These are very strong eggs, because they can be dropped multiple times without breaking as long as they are dropped from floors below their “threshold” floor, floor N. But once an egg is dropped from a floor above it’s threshold floor, it will break.

Output the minimum number of drops required to figure out N.

**Question – 93**

# Ants on a Triangle

There are three ants on a triangle, one at each corner.   
At a given moment in time, they all set off for a corner at random.   
What is the probability that they don’t collide?

*Answer is a floating point number.* Round it off to 2 decimal places and output it as I.xx where I is the integer part of the answer, and xx are 2 decimal digits after rounding off.  
For example, if the answer is 2/3, the response should be 0.67

**Question – 94**

# All Unique Permutations

**Problem Description**

Given an array **A** of size **N** denoting collection of numbers that might contain duplicates, return all possible unique permutations.

NOTE: No 2 entries in the permutation sequence should be the same.

**Input Format**

Only argument is an integer array A of size N.

**Output Format**

Return a 2-D array denoting all possible unique permutation of the array.

**Example Input**

Input 1:  A = [1, 1, 2] Input 2:  A = [1, 2]

**Example Output**

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

**Question – 95**

# Kth Permutation Sequence

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.

For example, given n = 3, k = 4, ans = "231"

**Good questions to ask the interviewer :**

* What if n is greater than 10. How should multiple digit numbers be represented in string?

In this case, just concatenate the number to the answer.  
so if n = 11, k = 1, ans = "1234567891011"

* Whats the maximum value of n and k?

In this case, k will be a positive integer thats less than INT\_MAX.  
n is reasonable enough to make sure the answer does not bloat up a lot.

**Question – 96**

# Combination Sum

**Problem Description**

Given an array of candidate numbers **A** and a target number **B**, find all unique combinations in A where the candidate numbers sums to B.

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

**Note**:

1) All numbers (including target) will be positive integers.

2) Elements in a combination (a1, a2, … , ak) must be in non-descending order. (ie, a1 ≤ a2 ≤ … ≤ ak).

3) The combinations themselves must be sorted in ascending order.

4) CombinationA > CombinationB iff (a1 > b1) OR (a1 = b1 AND a2 > b2) OR ... (a1 = b1 AND a2 = b2 AND ... ai = bi AND ai+1 > bi+1)

5) The solution set must not contain duplicate combinations.

**Problem Constraints**

1 <= |A| <= 20

1 <= A[i] <= 50

1 <= B <= 500

**Input Format**

The first argument is an integer array A.

The second argument is integer B.

**Output Format**

Return a vector of all combinations that sum up to B.

**Example Input**

Input 1:

A = [2, 3]

B = 2

Input 2:

A = [2, 3, 6, 7]

B = 7

**Example Output**

Output 1:

[ [2] ]

Output 2:

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

**Question – 97**

# Permutations

iven a collection of numbers, return all possible permutations.

**Example:**

[1,2,3] will have the following permutations:

[1,2,3]

[1,3,2]

[2,1,3]

[2,3,1]

[3,1,2]

[3,2,1]

**NOTE**

* No two entries in the permutation sequence should be the same.
* For the purpose of this problem, assume that all the numbers in the collection are unique.

**Warning :** DO NOT USE LIBRARY FUNCTION FOR GENERATING PERMUTATIONS.  
Example : next\_permutations in C++ / itertools.permutations in python.

**Question – 98**

# Generate all Parentheses II

Given n pairs of parentheses, write a function to generate all combinations of well-formed parentheses of length 2\*n.

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

"((()))", "(()())", "(())()", "()(())", "()()()"

Make sure the returned list of strings are sorted.

**Question – 99**

# Combination Sum II

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.
* Elements in a combination (a1, a2, … , ak) must be in non-descending order. (ie, a1 ≤ a2 ≤ … ≤ ak).
* The solution set must not contain duplicate combinations.

**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]

**Warning :** DO NOT USE LIBRARY FUNCTION FOR GENERATING COMBINATIONS.  
Example : itertools.combinations in python.

**Question – 100**

# NQueens

The n-queens puzzle is the problem of placing n queens on an n×n chessboard such that no two queens attack each other.



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.."]

]

**Question – 101**

# Combinations

Given two integers n and k, return all possible combinations of k numbers out of 1 2 3 ... n.

Make sure the combinations are **sorted**.

To elaborate,

1. Within every entry, elements should be sorted. [1, 4] is a valid entry while [4, 1] is not.
2. Entries should be sorted within themselves.

**Example :**  
If n = 4 and k = 2, a solution is:

[

[1,2],

[1,3],

[1,4],

[2,3],

[2,4],

[3,4],

]

**Warning :** DO NOT USE LIBRARY FUNCTION FOR GENERATING COMBINATIONS.

**Question – 102**

# Sudoku

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.

A picture containing text, crossword puzzle

Description automatically generated

A sudoku puzzle,

Text

Description automatically generated with low confidence

and its solution numbers marked in red.

**Example :**

For the above given diagrams, the corresponding input to your program will be

[[53..7....], [6..195...], [.98....6.], [8...6...3], [4..8.3..1], [7...2...6], [.6....28.], [...419..5], [....8..79]]

and we would expect your program to modify the above array of array of characters to

[[534678912], [672195348], [198342567], [859761423], [426853791], [713924856], [961537284], [287419635], [345286179]]

**Question – 103**

# Gray Code

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

There might be multiple gray code sequences possible for a given n.  
Return any such sequence.

**Question – 104**

# Subsets II

Given a collection of integers that might contain duplicates, S, return all possible subsets.

**Note:**

* Elements in a subset must be in non-descending order.
* The solution set must not contain duplicate subsets.
* The subsets must be sorted lexicographically.

**Example :**  
If S = [1,2,2], the solution is:

[

[],

[1],

[1,2],

[1,2,2],

[2],

[2, 2]

]

**Question – 105**

# Letter Phone

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.

A picture containing electronics, calculator, hand, orange

Description automatically generated

The digit 0 maps to 0 itself.  
The digit 1 maps to 1 itself.

Input: Digit string "23"

Output: ["ad", "ae", "af", "bd", "be", "bf", "cd", "ce", "cf"].

Make sure the returned strings are lexicographically sorted.

**Question – 106**

# Subset

Given a set of distinct integers, S, return all possible subsets.

**Note:**

* Elements in a subset must be in non-descending order.
* The solution set must not contain duplicate subsets.
* Also, the subsets should be sorted in ascending ( lexicographic ) order.
* The list is not necessarily sorted.

**Example :**

If S = [1,2,3], a solution is:

[

[],

[1],

[1, 2],

[1, 2, 3],

[1, 3],

[2],

[2, 3],

[3],

]

**Question – 107**

# Palindrome Partitioning

Given a string s, partition s such that every string of the partition is a palindrome.

Return all possible palindrome partitioning of s.

For example, given s = "aab",  
Return

[

["a","a","b"]

["aa","b"],

]

**Ordering the results in the answer :**

Entry i will come before Entry j if :

* len(Entryi[0]) < len(Entryj[0]) OR
* (len(Entryi[0]) == len(Entryj[0]) AND len(Entryi[1]) < len(Entryj[1])) OR  
  \*  
  \*  
  \*
* (len(Entryi[0]) == len(Entryj[0]) AND … len(Entryi[k] < len(Entryj[k]))

In the given example,  
["a", "a", "b"] comes before ["aa", "b"] because len("a") < len("aa")

**Question – 108**

# Substring Concatenation

You are given a string, S, and a list of words, L, that are all of the same length.

Find all starting indices of substring(s) in S that is a concatenation of each word in L exactly once and without any intervening characters.

**Example :**

S: "barfoothefoobarman"

L: ["foo", "bar"]

You should return the indices: [0,9].  
(order does not matter).

**Question – 109**

# Window String

Given a string S and a string T, find the minimum window in S which will contain all the characters in T in linear time complexity.  
Note that when the count of a character C in T is N, then the count of C in minimum window in S should be at least N.

**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, return the first occurring minimum window ( with minimum start index ).

**Question – 110**

# Longest Consecutive Sequence

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

**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.

**Question – 111**

# 4 Sum

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:**

* Elements in a quadruplet (a,b,c,d) must be in non-descending order. (ie, a ≤ b ≤ c ≤ d)
* The solution set must not contain duplicate quadruplets.

**Example :**   
Given array S = {1 0 -1 0 -2 2}, and target = 0  
A solution set is:

(-2, -1, 1, 2)

(-2, 0, 0, 2)

(-1, 0, 0, 1)

Also make sure that the solution set is lexicographically sorted.  
Solution[i] < Solution[j] iff Solution[i][0] < Solution[j][0] OR (Solution[i][0] == Solution[j][0] AND ... Solution[i][k] < Solution[j][k])

**Question – 112**

# Anagrams

Given an array of strings, return all groups of strings that are anagrams. Represent a group by a list of integers representing the index in the original list. Look at the sample case for clarification.

**Anagram :** a word, phrase, or name formed by rearranging the letters of another, such as 'spar', formed from 'rasp'

**Note:** All inputs will be in lower-case.

**Example :**

Input : cat dog god tca

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

cat and tca are anagrams which correspond to index 1 and 4.   
dog and god are another set of anagrams which correspond to index 2 and 3.  
The indices are 1 based ( the first element has index 1 instead of index 0).

**Ordering of the result :** You should not change the relative ordering of the words / phrases within the group. Within a group containing A[i] and A[j], A[i] comes before A[j] if i < j.

**Question – 113**

# Fraction

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.

**Example :**

Given numerator = 1, denominator = 2, return "0.5"

Given numerator = 2, denominator = 1, return "2"

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

**Question – 114**

# Points on the Straight Line

Given n points on a 2D plane, find the maximum number of points that lie on the same straight line.

Sample Input :

(1, 1)

(2, 2)

Sample Output :

2

You will be given 2 arrays X and Y. Each point is represented by (X[i], Y[i])

**Question – 115**

# 2 Sum

**Problem Description**

Given an array of integers, 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** < **index2**. Please note that your returned answers (both **index1** and **index2** ) are not zero-based. Put both these numbers in order in an array and return the array from your function ( Looking at the function signature will make things clearer ). Note that, if no pair exists, return empty list.

If multiple solutions exist, output the one where **index2** is minimum. If there are multiple solutions with the minimum **index2**, choose the one with minimum **index1** out of them. Input: [2, 7, 11, 15], target=9 Output: index1 = 1, index2 = 2

**Question – 116**

# Valid Sudoku

Determine if a Sudoku is valid, according to: http://sudoku.com.au/TheRules.aspx

The Sudoku board could be partially filled, where empty cells are filled with the character ‘.’.

A picture containing text, crossword puzzle

Description automatically generated

The input corresponding to the above configuration :

["53..7....", "6..195...", ".98....6.", "8...6...3", "4..8.3..1", "7...2...6", ".6....28.", "...419..5", "....8..79"]

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.

**Return 0 / 1 ( 0 for false, 1 for true ) for this problem**

**Question – 117**

# Copy List

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.

**Example**

Given list

1 -> 2 -> 3

with random pointers going from

1 -> 3

2 -> 1

3 -> 1

You should return a deep copy of the list. The returned answer should not contain the same node as the original list, but a copy of them. The pointers in the returned list should not link to any node in the original input list.

**Question – 118**

# Diffk II

Given an array A of integers and another non negative integer k, find if there exists 2 indices i and j such that A[i] - A[j] = k, i != j.

**Example :**

Input :

A : [1 5 3]

k : 2

Output :

1

as 3 - 1 = 2

* Return 0 / 1 for this problem.

**Question – 119**

# Longest Substring Without Repeat

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

**Example:**

The longest substring without repeating letters for "abcabcbb" is "abc", which the length is 3.

For "bbbbb" the longest substring is "b", with the length of 1.

**Question – 120**

# Diffk

Given an array ‘A’ of sorted integers and another non negative integer k, find if there exists 2 indices i and j such that A[i] - A[j] = k, i != j.

**Example:**

Input :

A : [1 3 5]

k : 4

Output : YES

as 5 - 1 = 4

Return 0 / 1 ( 0 for false, 1 for true ) for this problem

Try doing this in less than linear space complexity.

**Question – 121**

# Intersection Of Sorted Arrays

**Problem Description**

Find the intersection of two sorted arrays. OR in other words, Given 2 sorted arrays, find all the elements which occur in both the arrays.

**Example:**

Input:

A: [1 2 3 3 4 5 6]

B: [3 3 5]

Output: [3 3 5]

Input:

A: [1 2 3 3 4 5 6]

B: [3 5]

Output: [3 5]

**NOTE : For the purpose of this problem ( as also conveyed by the sample case ), assume that elements that appear more than once in both arrays should be included multiple times in the final output.**

**Question – 122**

# Merge Two Sorted Lists II

Given two sorted integer arrays A and B, merge B into A as one sorted array.

**Note**: You have to modify the array A to contain the merge of A and B. Do not output anything in your code.  
**TIP**: C users, please malloc the result into a new array and return the result.

If the number of elements initialized in A and B are m and n respectively, the resulting size of array A after your code is executed should be m + n

**Example :**

Input :

A : [1 5 8]

B : [6 9]

Modified A : [1 5 6 8 9]

**Question – 123**

# 3 Sum

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.

*Assume that there will only be one solution*

**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)

**Question – 124**

# Remove Duplicates from Sorted Array

**Problem Description**

Given a sorted array **A** consisting of duplicate elements.

Your task is to remove all the duplicates and return a sorted array of distinct elements consisting of all distinct elements present in **A**.

But, instead of returning an answer array, you have to **rearrange the given array in-place** such that it resembles what has been described above. Hence, return a single integer, the index(1-based) till which the answer array would reside in the given array **A**.

**Note**: This integer is the same as the number of integers remaining inside **A** had we removed all the duplicates. Look at the example explanations for better understanding.

**Input Format**

First and only argurment containing the integer array A.

**Output Format**

Return a single integer, as per the problem given.

**Example Input**

Input 1:

A = [1, 1, 2]

Input 2:

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

**Example Output**

Output 1:

2

Output 2:

3

**Question – 125**

# Sort by Color

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: Using library sort function is not allowed.*

**Example :**

Input : [0 1 2 0 1 2]

Modify array so that it becomes : [0 0 1 1 2 2]

**Question – 126**

# Array 3 Pointers

You are given 3 arrays A, B and C. All 3 of the arrays are sorted.

Find i, j, k such that :  
max(abs(A[i] - B[j]), abs(B[j] - C[k]), abs(C[k] - A[i])) is minimized.  
Return the minimum max(abs(A[i] - B[j]), abs(B[j] - C[k]), abs(C[k] - A[i]))

\*\*abs(x) is absolute value of x and is implemented in the following manner : \*\*

if (x < 0) return -x;

else return x;

**Example :**

Input :

A : [1, 4, 10]

B : [2, 15, 20]

C : [10, 12]

Output : 5

With 10 from A, 15 from B and 10 from C.

**Question – 127**

# Container With Most Water

Given **n non-negative integers a1, a2, ..., an,**  
where each represents a point at coordinate (i, ai).  
'n' vertical lines are drawn such that the two endpoints of line i is at (i, ai) and (i, 0).

Find two lines, which together with x-axis forms a container, such that the container contains the most water.

Your program should return an integer which corresponds to the maximum area of water that can be contained ( Yes, we know maximum area instead of maximum volume sounds weird. But this is 2D plane we are working with for simplicity ).

**Note:** You may not slant the container.

**Example :**

Input : [1, 5, 4, 3]

Output : 6

Explanation : 5 and 3 are distance 2 apart. So size of the base = 2. Height of container = min(5, 3) = 3.

So total area = 3 \* 2 = 6

**Question – 128**

# 3 Sum Zero

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:**

Elements in a triplet (a,b,c) must be in non-descending order. (ie, a ≤ b ≤ c)  
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)

**Question – 129**

# Remove Duplicates from Sorted Array II

**Remove Duplicates from Sorted Array**

Given a sorted array, remove the duplicates in place such that each element can appear atmost twice and return the new length.

Do not allocate extra space for another array, you must do this in place with constant memory.

**Note that even though we want you to return the new length, make sure to change the original array as well in place**

For example,  
Given input array A = [1,1,1,2],

Your function should return length = 3, and A is now [1,1,2].

**Question – 130**

# Remove Element from Array

**Remove Element**

Given an array and a value, remove all the instances of that value in the array.   
Also return the number of elements left in the array after the operation.  
It does not matter what is left beyond the expected length.

**Example:**  
If array A is [4, 1, 1, 2, 1, 3]  
and value elem is 1,   
then new length is 3, and A is now [4, 2, 3]

Try to do it in less than linear additional space complexity.

**Question – 131**

# Implement StrStr

**Problem Description**

Another question which belongs to the category of questions which are intentionally stated vaguely.  
  
Expectation is that you will ask for correct clarification or you will state your assumptions before you start coding.  
  
**Implement strStr().**  
strstr - locate a substring ( needle ) in a string ( haystack ).  
  
**Try not to use standard library string functions for this question.**  
  
Returns the index of the first occurrence of needle in haystack, or -1 if needle is not part of haystack.  
  
**NOTE:**String A is haystack, B is needle.  
  
**Good clarification questions:**

1. What should be the return value if the needle is empty?
2. What if both haystack and needle are empty?

For the purpose of this problem, assume that the return value should be -1 in both cases.

**Question – 132**

# Integer To Roman

Given an integer **A**, convert it to a roman numeral, and return a string corresponding to its roman numeral version

**Note :** This question has a lot of scope of clarification from the interviewer. Please take a moment to think of all the needed clarifications and see the expected response using “See Expected Output”

**Input Format**

The only argument given is integer A.

**Output Format**

Return a string denoting roman numeral version of A.

**Constraints**

1 <= A <= 3999

**For Example**

Input 1:

A = 5

Output 1:

"V"

Input 2:

A = 14

Output 2:

"XIV"

**Question – 133**

# Longest Common Prefix

**Problem Description**

Given the array of strings **A**, you need to find the longest string **S** which is the prefix of **ALL** the strings in the array.

Longest common prefix for a pair of strings **S1** and **S2** is the longest string **S** which is the prefix of both **S1** and **S2**.

**For Example:** longest common prefix of "abcdefgh" and "abcefgh" is "abc".

**Input Format**

The only argument given is an array of strings A.

**Output Format**

Return the longest common prefix of all strings in A.

**Example Input**

Input 1:

A = ["abcdefgh", "aefghijk", "abcefgh"]

Input 2:

A = ["abab", "ab", "abcd"];

**Example Output**

Output 1:

"a"

Output 2:

"ab"

**Question – 134**

# Roman To Integer

Given a string **A** representing a roman numeral.  
Convert **A** into integer.

**A** is guaranteed to be within the range from **1** to **3999**.

**NOTE**: Read more   
details about roman numerals at [Roman Numeric System](https://en.wikipedia.org/wiki/Roman_numerals#Roman_numeric_system)

**Input Format**

The only argument given is string A.

**Output Format**

Return an integer which is the integer verison of roman numeral string.

**For Example**

Input 1:

A = "XIV"

Output 1:

14

Input 2:

A = "XX"

Output 2:

20

**Question – 135**

# Length of Last Word

**Problem Description**

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:**

Given s = "Hello World",

return 5 as length("World") = 5.

Please make sure you try to solve this problem without using library functions. Make sure you only traverse the string once.

**Question – 136**

# Multiply Strings

Given two numbers represented as strings, return multiplication of the numbers as a string.

**Note:** The numbers can be arbitrarily large and are non-negative.  
**Note2:** Your answer should not have leading zeroes. For example, 00 is not a valid answer.

For example,   
given strings "12", "10", your answer should be “120”.

**NOTE** : DO NOT USE BIG INTEGER LIBRARIES ( WHICH ARE AVAILABLE IN JAVA / PYTHON ).

**Question – 137**

# Zigzag String

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"

\*\*Example 2 : \*\*  
ABCD, 2 can be written as

A....C

...B....D

and hence the answer would be ACBD.

**Question – 138**

# Atoi

Implement atoi to convert a string to an integer.

**Example :**

Input : "9 2704"

Output : 9

*Note: There might be multiple corner cases here. Clarify all your doubts using “See Expected Output”.*

**Questions:**

**Q1.** Does string contain whitespace characters before the number?  
**A.** Yes

**Q2.** Can the string have garbage characters after the number?  
**A.** Yes. Ignore it.

**Q3.** If no numeric character is found before encountering garbage characters, what should I do?  
**A.** Return 0.

**Q4.** What if the integer overflows?  
**A.** Return INT\_MAX if the number is positive, INT\_MIN otherwise.

**Warning : DO NOT USE LIBRARY FUNCTION FOR ATOI.**

**Question – 139**

# Valid Ip Addresses

Given a string containing only digits, restore it by returning all possible valid IP address combinations.

A valid IP address must be in the form of A.B.C.D, where A,B,C and D are numbers from 0-255. The numbers cannot be 0 prefixed unless they are 0.

**Example:**

Given “25525511135”,

return [“255.255.11.135”, “255.255.111.35”]. (Make sure the returned strings are sorted in order)

**Question – 140**

# Compare Version Numbers

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 < 1.13 < 1.13.4

**Question – 141**

# Longest Palindromic Substring

**Problem Description**

Given a string A of size N, find and return the **longest palindromic substring** in A.

Substring of string A is A[i...j] where 0 <= i <= j < len(A)

**Palindrome string:**  
A string which reads the same backwards. More formally, A is palindrome if reverse(A) = A.

**Incase of conflict**, return the substring which occurs first ( with the least starting index).

**Input Format**

First and only argument is a string A.

**Output Format**

Return a string denoting the longest palindromic substring of string A.

**Example Input**

A = "aaaabaaa"

**Example Output**

"aaabaaa"

**Question – 142**

# Pretty Json

Given a string **A** representating json object. Return an array of string denoting json object with proper indentaion.

Rules for proper indentaion:

* Every inner brace should increase one indentation to the following lines.
* Every close brace should decrease one indentation to the same line and the following lines.
* **The indents can be increased with an additional ‘\t’**

**Note**:

1. [] and {} are only acceptable braces in this case.
2. Assume for this problem that space characters can be done away with.

**Input Format**

The only argument given is the integer array A.

**Output Format**

Return a list of strings, where each entry corresponds to a single line. The strings should not have "\n" character in them.

**For Example**

Input 1:

A = "{A:"B",C:{D:"E",F:{G:"H",I:"J"}}}"

Output 1:

{

A:"B",

C:

{

D:"E",

F:

{

G:"H",

I:"J"

}

}

}

Input 2:

A = ["foo", {"bar":["baz",null,1.0,2]}]

Output 2:

[

"foo",

{

"bar":

[

"baz",

null,

1.0,

2

]

}

]

**Question – 143**

# Count And Say

**Problem Description**

The count-and-say sequence is the sequence of integers beginning as follows:  1, 11, 21, 1211, 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 sequence.

Note: The sequence of integers will be represented as a string.

**Example:**

if n = 2, the sequence is 11.

**Question – 144**

# Justified Text

**Problem Description**

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.

Your program should return a list of strings, where each string represents a single line.

**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.

**Question – 145**

# Add Binary Strings

**Problem Description**

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

**Example:**

a = "100"

b = "11"

Return a + b = "111".

**Question – 146**

# Reverse the String

Given a string **A**.

Return the string **A** after reversing the string word by word.

**NOTE**:

1. A sequence of non-space characters constitutes a word.
2. Your reversed string should not contain leading or trailing spaces, even if it is present in the input string.
3. If there are multiple spaces between words, reduce them to a single space in the reversed string.

**Input Format**

The only argument given is string A.

**Output Format**

Return the string A after reversing the string word by word.

**For Example**

Input 1:

A = "the sky is blue"

Output 1:

"blue is sky the"

Input 2:

A = "this is ib"

Output 2:

"ib is this"

**Question – 147**

# Reverse Bits

**Problem Description**

Reverse the bits of an **32** bit unsigned integer A.

**Problem Constraints**

0 <= A <= 232

**Input Format**

First and only argument of input contains an integer A.

**Output Format**

Return a single unsigned integer denoting the decimal value of reversed bits.

**Example Input**

Input 1:

0

Input 2:

3

**Example Output**

Output 1:

0

Output 2:

3221225472

**Question – 148**

# Number of 1 Bits

**Problem Description**

Write a function that takes an integer and returns the number of 1 bits it has.

**Problem Constraints**

1 <= A <= 109

**Input Format**

First and only argument contains integer A

**Output Format**

Return an integer as the answer

**Example Input**

**Input1:**

11

**Example Output**

**Output1:**

3

**Question – 149**

# Single Number

**Problem Description**

Given an array of integers **A**, 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?

**Problem Constraints**

* 2 <= **|A|** <= 2000000
* 0 <= A[i] <= INTMAX

**Input Format**

First and only argument of input contains an integer array A.

**Output Format**

Return a single integer denoting the single element.

**Example Input**

Input 1:

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

Input 2:

A = [1, 2, 2]

**Example Output**

Output 1:

3

Output 2:

1

**Question – 150**

# Divide Integers

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

Return the floor of the result of the division.

**Example:**

5 / 2 = 2

**Also, consider if there can be overflow cases. For overflow case, return INT\_MAX.**

Note: INT\_MAX = 2^31 - 1