Question 1

You are given a valid parentheses string — a string made only of `'('` and `')'` that forms correctly matched pairs.

A string is **valid** if:

- It's empty `""`, or
- It looks like `"(" + A + ")"` where `A` is valid, or
- It's made by joining two valid strings `A + B`.

Examples of valid strings:
`""`, `"()"`, `"(())()"`, `"(()(()))"`

A primitive valid parentheses string is a valid string that **cannot be split** into two smaller valid strings.
For example:

- `"()"` and `"(())"` are primitive
- `"(())()"` is **not** primitive (it can be split into `"(())"` and `"()"`)

Now, any valid string `s` can be broken into several primitive parts:

`s = P1 + P2 + ... + Pk`

For each primitive part, **remove its outermost pair of parentheses**, then join all the results together and return the new string.

Example 1:
Input: s = "(()())(())"
Output: "()()()"

Example 2:
Input: s = "(()())(())(()(()))"
Output: "()()()()(())"

Question 2

Given an input string s, reverse the order of the words.

A word is defined as a sequence of non-space characters. The words in s will be separated by at least one space.

Return a string of the words in reverse order concatenated by a single space.

Note that s may contain leading or trailing spaces or multiple spaces between two words. The returned string should only have a single space separating the words. Do not include any extra spaces.

Example 1:
Input: s = "the sky is blue"
Output: "blue is sky the"

Example 2:
Input: s = " hello world "
Output: "world hello"

Question 3

You are given a string num, representing a large integer. Return the largest-valued odd integer (as a string) that is a non-empty substring of num, or an empty string "" if no odd integer exists.

A substring is a contiguous sequence of characters within a string.

Example 1:

Input: num = "52"

Output: "5"

Example 2:

Input: num = "4206"

Output: ""

Question 4

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

If there is no common prefix, return an empty string "".

Example 1:

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

Output: "fl"

Example 2:

Input: strs = ["dog","racecar","car"]

Output: ""

Question 5

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

Two strings s and t 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.

Example 1:
Input: s = "egg", t = "add"
Output: true

Example 2:
Input: s = "foo", t = "bar"
Output: false

Question 6

Given two strings s and goal, return true if and only if s can become goal after some number of shifts on s.

A shift on s consists of moving the leftmost character of s to the rightmost position.

For example, if s = "abcde", then it will be "bcdea" after one shift.

Example 1:
Input: s = "abcde", goal = "cdeab"
Output: true

Example 2:
Input: s = "abcde", goal = "abced"
Output: false

Question 7

Given two strings s and t, return true if t is an anagram of s, and false otherwise.

Example 1:
Input: s = "anagram", t = "nagaram"
Output: true

Example 2:

Input: s = "rat", t = "car"

Output: false

Question 8

Given a string s, sort it in decreasing order based on the frequency of the characters. The frequency of a character is the number of times it appears in the string.

Return the sorted string. If there are multiple answers, return any of them.

Example 1:

Input: s = "tree"

Output: "eert"

Example 2:

Input: s = "cccaaa"

Output: "aaaccc"

Question 9

Given a valid parentheses string s, return the nesting depth of s. The nesting depth is the maximum number of nested parentheses.

Example 1:

Input: s = "(1+(2*3)+((8)/4))+1"

Output: 3

Example 2:

Input: s = "(1)+((2))+(((3)))"

Output: 3

Question 10

Roman numerals are represented by seven different symbols: I, V, X, L, C, D and M.

Symbol	**Value**
I	1
V	5
X	10
L	50
C	100
D	500
M	1000

For example, 2 is written as II in Roman numeral, just two ones added together. 12 is written as XII, which is simply X + II. The number 27 is written as XXVII, which is XX + V + II.

Roman numerals are usually written largest to smallest from left to right. However, the numeral for four is not IIII. Instead, the number four is written as IV. Because the one is before the five we subtract it making four. The same principle applies to the number nine, which is written as IX. There are six instances where subtraction is used:

- I can be placed before V (5) and X (10) to make 4 and 9.
- X can be placed before L (50) and C (100) to make 40 and 90.
- C can be placed before D (500) and M (1000) to make 400 and 900.

Given a roman numeral, convert it to an integer.

Example 1:

Input: s = "III"
Output: 3

Example 2:

Input: s = "LVIII"
Output: 58

Question 11

Implement the `myAtoi(string s)` function, which converts a string to a 32-bit signed integer.

The algorithm for `myAtoi(string s)` is as follows:

1. **Whitespace:** Ignore any leading whitespace (" ").

2. **Signedness:** Determine the sign by checking if the next character is '-' or '+', assuming positivity if neither present.

3. **Conversion:** Read the integer by skipping leading zeros until a non-digit character is encountered or the end of the string is reached. If no digits were read, then the result is 0.

4. **Rounding:** If the integer is out of the 32-bit signed integer range [-2^{31}, $2^{31} - 1$], then round the integer to remain in the range. Specifically, integers less than -2^{31} should be rounded to -2^{31}, and integers greater than $2^{31} - 1$ should be rounded to $2^{31} - 1$.

Return the integer as the final result.

Example 1:
Input: s = "42"
Output: 42

Example 2:
Input: s = " -042"
Output: -42

Question 12

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

Letters are case sensitive, for example, "Aa" is not considered a palindrome.

Example 1:
Input: s = "abccccdd"
Output: 7

Example 2:
Input: s = "a"
Output: 1

Question 13

Given a string array `words`, return an array of all characters that show up in all strings within the `words` (including duplicates). You may return the answer in any order.

Example 1:

Input: words = ["bella","label","roller"]

Output: ["e","l","l"]

Example 2:

Input: words = ["cool","lock","cook"]

Output: ["c","o"]

Question 14

Given a string s of lower and upper case English letters.

A good string is a string which doesn't have two adjacent characters `s[i]` and `s[i + 1]` where:

- 0 <= i <= s.length - 2
- `s[i]` is a lower-case letter and `s[i + 1]` is the same letter but in upper-case or vice-versa.

To make the string good, you can choose two adjacent characters that make the string bad and remove them. You can keep doing this until the string becomes good.

Return the string after making it good. The answer is guaranteed to be unique under the given constraints.

Notice that an empty string is also good.

Example 1:

Input: s = "leEeetcode"

Output: "leetcode"

Example 2:

Input: s = "abBAcC"

Output: ""

Question 15

You are given two strings `word1` and `word2`. Merge the strings by adding letters in alternating order, starting with `word1`. If a string is longer than the other, append the additional letters onto the end of the merged string.

Return the merged string.

Example 1:

Input: word1 = "abc", word2 = "pqr"

Output: "apbqcr"

Example 2:

Input: word1 = "ab", word2 = "pqrs"

Output: "apbqrs"

Question 16

Given an array `nums` containing n distinct numbers in the range [0, n], return the only number in the range that is missing from the array.

Example 1:

Input: nums = [3,0,1]

Output: 2

Example 2:

Input: nums = [0,1]

Output: 2

Question 17

Given an integer array `nums`, move all 0's to the end of it while maintaining the relative order of the non-zero elements.

Note that you must do this in-place without making a copy of the array.

Example 1:

Input: nums = [0,1,0,3,12]

Output: [1,3,12,0,0]

Example 2:

Input: nums = [0]

Output: [0]

Question 18

Given a binary array `nums`, return the maximum number of consecutive 1's in the array.

Example 1:

Input: nums = [1,1,0,1,1,1]

Output: 3

Example 2:

Input: nums = [1,0,1,1,0,1]

Output: 2

Question 19

Given a non-empty array of integers `nums`, every element appears twice except for one. Find that single one.

Example 1:

Input: nums = [2,2,1]

Output: 1

Example 2:

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

Output: 4

Question 20

Given an array `nums` of size n, return the majority element.

The majority element is the element that appears more than ⌊n / 2⌋ times. You may assume that the majority element always exists in the array.

Example 1:

Input: nums = [3,2,3]

Output: 3

Example 2:

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

Output: 2

Question 21

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

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

You can return the answer in any order.

Example 1:
Input: nums = [2,7,11,15], target = 9
Output: [0,1]

Example 2:
Input: nums = [3,2,4], target = 6
Output: [1,2]

Question 22

You are given an array `prices` where `prices[i]` is the price of a given stock on the ith day.

You want to maximize your profit by choosing a single day to buy one stock and choosing a different day in the future to sell that stock.

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

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

Example 2:

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

Output: 0

Question 23

You are given a 0-indexed integer array `nums` of even length consisting of an equal number of positive and negative integers.

You should return the array of `nums` such that the array follows the given conditions:

1. Every consecutive pair of integers have opposite signs.
2. For all integers with the same sign, the order in which they were present in `nums` is preserved.
3. The rearranged array begins with a positive integer.

Return the modified array after rearranging the elements to satisfy the aforementioned conditions.

Example 1:

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

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

Example 2:

Input: nums = [-1,1]

Output: [1,-1]

Question 24

A permutation of an array of integers is an arrangement of its members into a sequence or linear order.

The next permutation of an array of integers is the next lexicographically greater permutation of its integers. More formally, if all the permutations of the array are sorted in one container according to their lexicographical order, then the next permutation of that array is the permutation that follows it in the sorted container. If such arrangement is not possible, the array must be rearranged as the lowest possible order (i.e., sorted in ascending order).

Given an array of integers `nums`, find the next permutation of `nums`.

Example 1:

Input: nums = [1,2,3]

Output: [1,3,2]

Example 2:

Input: nums = [3,2,1]

Output: [1,2,3]

Question 25

Given an array of integers `nums` and an integer `k`, return the total number of subarrays whose sum equals to `k`.

A subarray is a contiguous non-empty sequence of elements within an array.

Example 1:

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

Output: 2

Example 2:

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

Output: 2

Question 26

Given an m x n integer matrix `matrix`, if an element is 0, set its entire row and column to 0's.

Example 1:
Input: matrix = [[1,1,1],[1,0,1],[1,1,1]]
Output: [[1,0,1],[0,0,0],[1,0,1]]

Example 2:
Input: matrix = [[0,1,2,0],[3,4,5,2],[1,3,1,5]]
Output: [[0,0,0,0],[0,4,5,0],[0,3,1,0]]

Question 27

You are given an n x n 2D matrix representing an image. Rotate the image by 90 degrees (clockwise).

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:
Input: matrix = [[1,2,3],[4,5,6],[7,8,9]]
Output: [[7,4,1],[8,5,2],[9,6,3]]

Example 2:
Input: matrix = [[5,1,9,11],[2,4,8,10],[13,3,6,7],[15,14,12,16]]
Output: [[15,13,2,5],[14,3,4,1],[12,6,8,9],[16,7,10,11]]

Question 28

Given an m x n matrix, return all elements of the matrix in spiral order.

Example 1:

Input: matrix = [[1,2,3],[4,5,6],[7,8,9]]

Output: [1,2,3,6,9,8,7,4,5]

Example 2:

Input: matrix = [[1,2,3,4],[5,6,7,8],[9,10,11,12]]

Output: [1,2,3,4,8,12,11,10,9,5,6,7]

Question 29

You are given a `row x col` grid representing a map where `grid[i][j] = 1` represents land and `grid[i][j] = 0` represents water.

Grid cells are connected horizontally/vertically (not diagonally). The grid is completely surrounded by water, and there is exactly one island (i.e., one or more connected land cells).

The island doesn't have "lakes", meaning the water inside isn't connected to the water around the island. One cell is a square with side length 1. The grid is rectangular, width and height don't exceed 100. Determine the perimeter of the island.

Example 1:

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

Output: 16

Example 2:

Input: grid = [[1]]

Output: 4

Question 30

Given an m x n matrix, return true if the matrix is Toeplitz. Otherwise, return false.

A matrix is Toeplitz if every diagonal from top-left to bottom-right has the same elements.

Example 1:

Input: matrix = [[1,2,3,4],[5,1,2,3],[9,5,1,2]]

Output: true

Example 2:

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

Output: false