# 大厂算法面试题库中高频出现的 30 道典型题

算法是程序员求职面试时绕不开的坎，我们从大厂算法面试题库中抽取了典型的 30 道题，难度跨越了简单、中等和困难，大家可以通过完成以下 30 道题来检测自己的算法能力。

1、给定一个字符串，找出不含有重复字符的**最长子串**的长度。

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

2、给定一个 32 位有符号整数，将整数中的数字进行反转。

Given a 32-bit signed integer, reverse digits of an integer.

3、实现 atoi，将字符串转为整数。

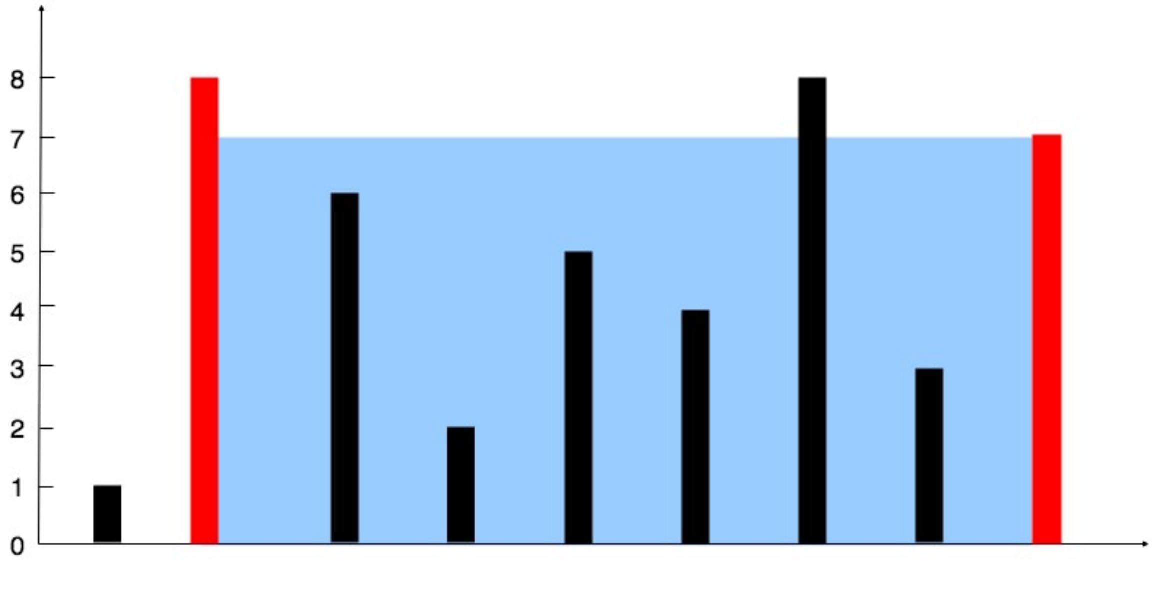
Implement atoi which converts a string to an integer.

4、给定 *n* 个非负整数 *a*1，*a*2，...，*a*n，每个数代表坐标中的一个点 (*i*, *ai*) 。在坐标内画 *n* 条垂直线，垂直线 *i* 的两个端点分别为 (*i*, *ai*) 和 (*i*, 0)。找出其中的两条线，使得它们与 *x* 轴共同构成的容器可以容纳最多的水。

**说明：**你不能倾斜容器，且 *n* 的值至少为 2。

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.

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



图中垂直线代表输入数组 [1,8,6,2,5,4,8,3,7]。在此情况下，容器能够容纳水（表示为蓝色部分）的最大值为 49。

The above vertical lines are represented by array [1,8,6,2,5,4,8,3,7]. In this case, the max area of water (blue section) the container can contain is 49.

5、编写一个函数来查找字符串数组中的最长公共前缀。

如果不存在公共前缀，返回空字符串 ""。

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 "".

6、给定一个排序数组和一个目标值，在数组中找到目标值，并返回其索引。如果目标值不存在于数组中，返回它将会被按顺序插入的位置。

你可以假设数组中无重复元素。

Given a sorted array and a target value, return the index if the target is found. If not, return the index where it would be if it were inserted in order.

You may assume no duplicates in the array.

7、给定一个非负整数数组，你最初位于数组的第一个位置。

数组中的每个元素代表你在该位置可以跳跃的最大长度。

你的目标是使用最少的跳跃次数到达数组的最后一个位置。

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

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

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

8、给定一个 $n \times n$ 的二维矩阵表示一个图像。

将图像顺时针旋转 90 度。

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

Rotate the image by 90 degrees (clockwise).

9、给定一个二叉树，返回其按层次遍历的节点值。（即逐层地，从左到右访问所有节点）。

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

10、给定一个**非空**整数数组，除了某个元素只出现一次以外，其余每个元素均出现两次。找出那个只出现了一次的元素。

**说明：**

你的算法应该具有线性时间复杂度。 你可以不使用额外空间来实现吗？

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

**Note:**

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

11、给定一个**非空**整数数组，除了某个元素只出现一次以外，其余每个元素均出现了三次。找出那个只出现了一次的元素。

Given a **non-empty** array of integers, every element appears *three* times except for one, which appears exactly once. Find that single one.

12、在 $O(n \log n)$ 时间复杂度和常数级空间复杂度下，对链表进行排序。

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

13、峰值元素是指其值大于左右相邻值的元素。

给定一个输入数组 nums，其中 nums[i] ≠ nums[i+1]，找到峰值元素并返回其索引。

数组可能包含多个峰值，在这种情况下，返回任何一个峰值所在位置即可。

你可以假设 nums[-1] = nums[n] = -∞。

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

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

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

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

14、给定一个无序的数组，找出数组在排序之后，相邻元素之间最大的差值。

如果数组元素个数小于 2，则返回 0。

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

Return 0 if the array contains less than 2 elements.

15、给定两个整数，分别表示分数的分子 numerator 和分母 denominator，以字符串形式返回小数。

如果小数部分为循环小数，则将循环的部分括在括号内。

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.

16、实现一个 Trie (前缀树)，包含 insert, search, 和 startsWith 这三个操作。

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

17、给定一个**不含重复**单词的列表，编写一个程序，返回给定单词列表中所有的连接词。

连接词的定义为：一个字符串完全是由至少两个给定数组中的单词组成的。

Given a list of words (**without duplicates**), please write a program that returns all concatenated words in the given list of words.

A concatenated word is defined as a string that is comprised entirely of at least two shorter words in the given array.

18、中位数是有序序列最中间的那个数。如果序列的大小是偶数，则没有最中间的数；此时中位数是最中间的两个数的平均数。

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

19、回忆一下祖玛游戏。现在桌上有一串球，颜色有红色(R)，黄色(Y)，蓝色(B)，绿色(G)，还有白色(W)。 现在你手里也有几个球。

每一次，你可以从手里的球选一个，然后把这个球插入到一串球中的某个位置上（包括最左端，最右端）。接着，如果有出现三个或者三个以上颜色相同的球相连的话，就把它们移除掉。重复这一步骤直到桌上所有的球都被移除。

找到插入并可以移除掉桌上所有球所需的最少的球数。如果不能移除桌上所有的球，输出 -1 。

Think about Zuma Game. You have a row of balls on the table, colored red(R), yellow(Y), blue(B), green(G), and white(W). You also have several balls in your hand.

Each time, you may choose a ball in your hand, and insert it into the row (including the leftmost place and rightmost place). Then, if there is a group of 3 or more balls in the same color touching, remove these balls. Keep doing this until no more balls can be removed.

Find the minimal balls you have to insert to remove all the balls on the table. If you cannot remove all the balls, output -1.

20、一个 N x N 的网格(grid) 代表了一块樱桃地，每个格子由以下三种数字的一种来表示：

- 0 表示这个格子是空的，所以你可以穿过它。
- 1 表示这个格子里装着一个樱桃，你可以摘到樱桃然后穿过它。
- -1 表示这个格子里有荆棘，挡着你的路。

你的任务是在遵守下列规则的情况下，尽可能的摘到最多樱桃：

- 从位置 (0, 0) 出发，最后到达 (N-1, N-1) ，只能向下或向右走，并且只能穿越有效的格子（即只可以穿过值为 0 或者 1 的格子）；
- 当到达 (N-1, N-1) 后，你要继续走，直到返回到 (0, 0) ，只能向上或向左走，并且只能穿越有效的格子；
- 当你经过一个格子且这个格子包含一个樱桃时，你将摘到樱桃并且这个格子会变成空的（值变为 0）；
- 如果在 (0, 0) 和 (N-1, N-1) 之间不存在一条可经过的路径，则没有任何一个樱桃能被摘到。

In a N x N grid representing a field of cherries, each cell is one of three possible integers.

0 means the cell is empty, so you can pass through;

1 means the cell contains a cherry, that you can pick up and pass through;

-1 means the cell contains a thorn that blocks your way.

Your task is to collect maximum number of cherries possible by following the rules below:

Starting at the position (0, 0) and reaching (N-1, N-1) by moving right or down through valid path cells (cells with value 0 or 1);

After reaching (N-1, N-1), returning to (0, 0) by moving left or up through valid path cells;

When passing through a path cell containing a cherry, you pick it up and the cell becomes an empty cell (0);

If there is no valid path between (0, 0) and (N-1, N-1), then no cherries can be collected.

21、*自除数* 是指可以被它包含的每一位数除尽的数。

例如，128 是一个自除数，因为 128 % 1 == 0，128 % 2 == 0，128 % 8 == 0。

还有，自除数不允许包含 0 。

给定上边界和下边界数字，输出一个列表，列表的元素是边界（含边界）内所有的自除数。

A *self-dividing number* is a number that is divisible by every digit it contains.

For example, 128 is a self-dividing number because 128 % 1 == 0, 128 % 2 == 0, and 128 % 8 == 0.

Also, a self-dividing number is not allowed to contain the digit zero.

Given a lower and upper number bound, output a list of every possible self dividing number, including the bounds if possible.

22、给定一个字符串 S，找出 S 中不同的非空回文子字符串个数，并**返回该数字与 10^9 + 7 的模。**

通过从 S 中删除 0 个或多个字符来获得子字符串。

如果一个字符串字符序列与它的反转字符串字符序列一致，那么它是回文字符串。

如果对于某个 i， A_i != B_i，那么 A_1, A_2, ... 和 B_1, B_2, ... 这两个字符串是不同的。

Given a string S, find the number of different non-empty palindromic subsequences in S, and **return that number modulo 10^9 + 7.**

A subsequence of a string S is obtained by deleting 0 or more characters from S.

A sequence is palindromic if it is equal to the sequence reversed.

Two sequences A_1, A_2, ... and B_1, B_2, ... are different if there is some i for which A_i != B_i.

23、实现一个 MyCalendar 类来存放你的日程安排，你可以一直添加新的日程安排。

MyCalendar 有一个 book(int start, int end)方法。它意味着在 start 到 end 时间内增加一个日程安排，注意，这里的时间是半开区间，即 [start, end), 实数 x 的范围为， start <= x < end。

当 **K** 个日程安排有一些时间上的交叉时（例如 K 个日程安排都在同一时间内），就会产生 **K** 次预订。

每次调用 MyCalendar.book 方法时，返回一个整数 K ，表示最大的 K 次预订。

请按照以下步骤调用 MyCalendar 类: MyCalendar cal = new MyCalendar();MyCalendar.book(start, end)

Implement a MyCalendarThree class to store your events. A new event can **always** be added.

Your class will have one method, book(int start, int end). Formally, this represents a booking on the half open interval [start, end), the range of real numbers x such that start <= x < end.

A *K-booking* happens when **K** events have some non-empty intersection (ie., there is some time that is common to all K events.)

For each call to the method MyCalendar.book, return an integer Krepresenting the largest integer such that there exists a K-booking in the calendar.

Your class will be called like this:MyCalendarThree cal = new MyCalendarThree();MyCalendarThree.book(start, end)

24、你有一个带有四个圆形拨轮的转盘锁。每个拨轮都有 10 个数字： '0', '1', '2', '3', '4', '5', '6', '7', '8', '9' 。每个拨轮可以自由旋转：例如把 '9'变为 '0'，'0' 变为 '9' 。每次旋转都只能旋转一个拨轮的一位数字。

锁的初始数字为 '0000' ，一个代表四个拨轮的数字的字符串。

列表 deadends 包含了一组死亡数字，一旦拨轮的数字和列表里的任何一个元素相同，这个锁将会被永久锁定，无法再被旋转。

字符串 target 代表可以解锁的数字，你需要给出最小的旋转次数，如果无论如何不能解锁，返回 -1。

You have a lock in front of you with 4 circular wheels. Each wheel has 10 slots: '0', '1', '2', '3', '4', '5', '6', '7', '8', '9'. The wheels can rotate freely and wrap around: for example we can turn '9' to be '0', or '0' to be '9'. Each move consists of turning one wheel one slot.

The lock initially starts at '0000', a string representing the state of the 4 wheels.

You are given a list of deadends dead ends, meaning if the lock displays any of these codes, the wheels of the lock will stop turning and you will be unable to open it.

Given a target representing the value of the wheels that will unlock the lock, return the minimum total number of turns required to open the lock, or -1 if it is impossible.

25、您将获得一个双向链表，除了下一个和前一个指针之外，它还有一个子指针，可能指向单独的双向链表。这些子列表可能有一个或多个自己的子项，依此类推，生成多级数据结构，如下面的示例所示。

扁平化列表，使所有结点出现在单级双链表中。您将获得列表第一级的头部。

You are given a doubly linked list which in addition to the next and previous pointers, it could have a child pointer, which may or may not point to a separate doubly linked list. These child lists may have one or more children of their own, and so on, to produce a multilevel data structure, as shown in the example below.

Flatten the list so that all the nodes appear in a single-level, doubly linked list. You are given the head of the first level of the list.

26、给定两个整数 L 和 R ，找到闭区间 [L, R] 范围内，计算置位位数为质数的整数个数。

（注意，计算置位代表二进制表示中 1 的个数。例如 21 的二进制表示 10101 有 3 个计算置位。还有，1 不是质数。）

Given two integers L and R, find the count of numbers in the range [L, R](inclusive) having a prime number of set bits in their binary representation.

(Recall that the number of set bits an integer has is the number of 1s present when written in binary. For example, 21written in binary is 10101 which has 3 set bits. Also, 1 is not a prime.)

27、给定一个字符串 S，检查是否能重新排布其中的字母，使得两相邻的字符不同。

若可行，输出任意可行的结果。若不可行，返回空字符串。

Given a string S, check if the letters can be rearranged so that two characters that are adjacent to each other are not the same.

If possible, output any possible result. If not possible, return the empty string.

28、机器人在一个无限大小的网格上行走，从点 (0, 0) 处开始出发，面向北方。该机器人可以接收以下三种类型的命令：

- -2：向左转 90 度

- -1：向右转 90 度

- 1 <= x <= 9：向前移动 x 个单位长度

在网格上有一些格子被视为障碍物。

第 i 个障碍物位于网格点 (obstacles[i][0], obstacles[i][1])

如果机器人试图走到障碍物上方，那么它将停留在障碍物的前一个网格方块上，但仍然可以继续该路线的其余部分。

返回从原点到机器人的最大欧式距离的**平方**。

A robot on an infinite grid starts at point (0, 0) and faces north. The robot can receive one of three possible types of commands:

- -2: turn left 90 degrees

- -1: turn right 90 degrees

- 1 <= x <= 9: move forward xunits

Some of the grid squares are obstacles.

The i-th obstacle is at grid point (obstacles[i][0], obstacles[i][1])

If the robot would try to move onto them, the robot stays on the previous grid square instead (but still continues following the rest of the route.)

Return the **square** of the maximum Euclidean distance that the robot will be from the origin.

29、给定两个句子 A 和 B 。（*句子*是一串由空格分隔的单词。每个*单词*仅由小写字母组成。）

如果一个单词在其中一个句子中只出现一次，在另一个句子中却没有出现，那么这个单词就是*不常见的*。

返回所有不常用单词的列表。

您可以按任何顺序返回列表。

We are given two sentences A and B. (A *sentence* is a string of space separated words. Each *word* consists only of lowercase letters.)

A word is *uncommon* if it appears exactly once in one of the sentences, and does not appear in the other sentence.

Return a list of all uncommon words.

You may return the list in any order.

30、你的音乐播放器里有 N 首不同的歌，在旅途中，你的旅伴想要听 L 首歌（不一定不同，即，允许歌曲重复）。请你为她按如下规则创建一个播放列表：

- 每首歌至少播放一次。
- 一首歌只有在其他 K 首歌播放完之后才能再次播放。

返回可以满足要求的播放列表的数量。**由于答案可能非常大，请返回它模 10^9 + 7 的结果。**

Your music player contains N different songs and she wants to listen to L (not necessarily different) songs during your trip. You create a playlist so that:

- Every song is played at least once

- A song can only be played again only if K other songs have been played

Return the number of possible playlists.  **As the answer can be very large, return it modulo 10^9 + 7**.

如果你想进入硅谷的一线互联网公司如 Facebook、Google 等，或者想进入国内的一线大厂如腾讯、阿里、百度、今日头条等，极客时间《算法面试通关》40 节视频课程将是你很好的选择。那怕你暂时不需要面试，也可以通过这个课程中的实战演练来提升算法能力。

作者覃超，曾在 Facebook 做工程师，作为 Facebook Messenger Tech Lead，参与了 Facebook App、Facebook Messenger、Facebook Phone 等产品的研发工作。在 Facebook 的 3 年时间，面试过数百位技术人，对于算法面试的考察点和应对方式有丰富的第一手经验。相信只要你认真看完这门课程，一定会对算法面试更有把握，提升通关率。

扫码直通课程主页——