# 题目

给你一个包含若干星号\*的字符串s。

在一步操作中，你可以：

选中s中的一个星号。

移除星号左侧最近的那个非星号字符，并移除该星号自身。

返回移除所有星号之后的字符串。

注意：

生成的输入保证总是可以执行题面中描述的操作。

可以证明结果字符串是唯一的。

示例 1：

输入：s = "leet\*\*cod\*e"

输出："lecoe"

解释：从左到右执行移除操作：

- 距离第 1 个星号最近的字符是 "leet\*\*cod\*e" 中的 't' ，s 变为 "lee\*cod\*e" 。

- 距离第 2 个星号最近的字符是 "lee\*cod\*e" 中的 'e' ，s 变为 "lecod\*e" 。

- 距离第 3 个星号最近的字符是 "lecod\*e" 中的 'd' ，s 变为 "lecoe" 。

不存在其他星号，返回 "lecoe" 。

示例 2：

输入：s = "erase\*\*\*\*\*"

输出：""

解释：整个字符串都会被移除，所以返回空字符串。

提示：

1 <= s.length <= 105

s 由小写英文字母和星号 \* 组成

s 可以执行上述操作

# 分析

## 方法一：模拟

class Solution {

public:

string removeStars(string s) {

string ans;

for (auto c : s) {

if (c != '\*') {

ans += c;

} else {

ans.pop\_back();

}

}

return ans;

}

};

## 方法二：栈