# 题目

返回 s 字典序最小的子序列，该子序列包含 s 的所有不同字符，且只包含一次。

示例 1：

输入：s = "bcabc"

输出："abc"

示例 2：

输入：s = "cbacdcbc"

输出："acdb"

提示：

1 <= s.length <= 1000

s 由小写英文字母组成

注意：该题与 316 https://leetcode.cn/problems/remove-duplicate-letters/ 相同

# 分析

class Solution {

public:

string smallestSubsequence(string s) {

vector<int> left(26), in\_ans(26);

for (char c: s)

left[c - 'a']++; // 统计每个字母的出现次数

string ans;

for (char c: s) {

left[c - 'a']--;

if (in\_ans[c - 'a']) // ans 中不能有重复字母

continue;

while (!ans.empty() && c < ans.back() && left[ans.back() - 'a']) {

// (设 x=ans.back()) 如果 c < x，且右边还有 x，那么可以把 x 去掉，

// 因为后面可以重新把 x 加到 ans 中

in\_ans[ans.back() - 'a'] = false; // 标记 x 不在 ans 中

ans.pop\_back();

}

ans += c; // 把 c 加到 ans 的末尾

in\_ans[c - 'a'] = true; // 标记 c 在 ans 中

}

return ans;

}

};