# 题目

字符串压缩。利用字符重复出现的次数，编写一种方法，实现基本的字符串压缩功能。比如，字符串aabcccccaaa会变为a2b1c5a3。若“压缩”后的字符串没有变短，则返回原先的字符串。你可以假设字符串中只包含大小写英文字母（a至z）。

示例1:

输入："aabcccccaaa"

输出："a2b1c5a3"

示例2:

输入："abbccd"

输出："abbccd"

解释："abbccd"压缩后为"a1b2c2d1"，比原字符串长度更长。

提示：

字符串长度在[0, 50000]范围内。

# 分析

## 方法一：模拟

思路：

代码：

class Solution {

public:

std::string compressString(std::string S) {

if (S.empty()) {

return S;

}

std::string compressed;

char currentChar = S[0];

int count = 1;

for (int i = 1; i < S.length(); ++i) {

if (S[i] == currentChar) {

++count;

} else {

compressed += currentChar;

compressed += std::to\_string(count);

currentChar = S[i];

count = 1;

}

}

// Add the last character and its count

compressed += currentChar;

compressed += std::to\_string(count);

// Compare the lengths and return the shorter string

return compressed.length() < S.length() ? compressed : S;

}

};