

250715反转字符串中的单词（题解）

题目链接：

<https://leetcode.cn/problems/reverse-words-in-a-string/description/>

给你一个字符串 `s`，请你反转字符串中 单词 的顺序。

单词 是由非空格字符组成的字符串。`s` 中使用至少一个空格将字符串中的 单词 分隔开。

返回 单词 顺序颠倒且 单词 之间用单个空格连接的结果字符串。

注意：输入字符串 `s` 中可能会存在前导空格、尾随空格或者单词间的多个空格。返回的结果字符串中，单词间应当仅用单个空格分隔，且不包含任何额外的空格。

示例 1：

输入：`s = "the sky is blue"`

输出：`"blue is sky the"`

示例 2：

输入：`s = " hello world "`

输出：`"world hello"`

解释：反转后的字符串中不能存在前导空格和尾随空格。

示例 3：

输入：`s = "a good example"`

输出：`"example good a"`

解释：如果两个单词间有多余的空格，反转后的字符串需要将单词间的空格减少到仅有一个。

解法一：双指针

1. 初始化指针：

- 从字符串末尾 `m = s.size() - 1` 开始向前遍历。
- 跳过结尾的空格。

2. 双指针提取单词：

- `n` 记录当前单词的尾部位置。
- `m` 往前移动，直到遇到空格，表示找到了一个完整单词。
- 使用 `s.substr(m + 1, n - m)` 提取单词，加到结果字符串中。

3. 跳过单词间多余空格，更新 `n = m`，继续提取下一个单词。

4. 去除末尾多余空格：最终结果可能多一个尾部空格，使用 `res.substr(0, res.size() - 1)` 去掉。

```

class Solution {
public:
    string reverseWords(string s) {
        // 使用双指针
        int m = s.size() - 1;
        string res;
        // 除去尾部空格
        while (m >= 0 && s[m] == ' ') m--;
        int n = m; // n是当前单词的结尾位置
        while (m >= 0) {
            // 找到单词开头
            while (m >= 0 && s[m] != ' ') m--;
            // 提取单词并加到结果中（带一个空格）
            res += s.substr(m + 1, n - m) + " ";
            // 跳过多个空格
            while (m >= 0 && s[m] == ' ') m--;
            // 更新 n 为下一个单词的结尾
            n = m;
        }
        // 去掉结果末尾多余空格
        return res.substr(0, res.size() - 1);
    }
};

```

示例

输入：

```
"  a   good   example  "
```

处理步骤：

1. 去尾部空格后开始倒序扫描。
2. 提取 "example"、"good"、"a"。
3. 拼接为 "example good a "。
4. 去除尾部空格，返回 "example good a"。