# 题目

输入一个英文句子，翻转句子中单词的顺序，但单词内字符的顺序不变。为简单起见，标点符号和普通字母一样处理。例如输入字符串"I am a student. "，则输出"student. a am I"。

**示例 1：**

输入: "the sky is blue"

输出: "blue is sky the"

**示例 2：**

输入: "  hello world!  "

输出: "world! hello"

解释: 输入字符串可以在前面或者后面包含多余的空格，但是反转后的字符不能包括。

**示例 3：**

输入: "a good   example"

输出: "example good a"

解释: 如果两个单词间有多余的空格，将反转后单词间的空格减少到只含一个。

**说明：**

无空格字符构成一个单词。

输入字符串可以在前面或者后面包含多余的空格，但是反转后的字符不能包括。

如果两个单词间有多余的空格，将反转后单词间的空格减少到只含一个。

**注意：**本题与主站 151 题相同：

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

注意：此题对比原题有改动

# 分析

## 方法一：栈+istringstream

class Solution {

public:

string reverseWords(string s) {

stack<string> stk;

string ret;

string tmp;

istringstream ss(s);

while(ss>>tmp)

{

stk.push(tmp);

stk.push(" ");

}

//去掉一个多余的空格

if(!stk.empty())

stk.pop();

while(!stk.empty())

{

ret += stk.top();

stk.pop();

}

return ret;

}

};

## 方法二：istringstream

代码：

class Solution {

public:

string reverseWords(string s) {

istringstream ss(s);

string res, str;

while(ss >> str)

res = str + ' ' + res;

return res.substr(0, res.size() - 1);

}

};

## 方法三：双指针

**解题思路：**

1、我们使用两个指针l、r来帮助选择每一个单词

2、在每次循环时，先去除所有单词右侧空格，获取某个单词的最右下标r，再获取单词的最左下标l

3、然后把单词s.substr(l + 1, r - l)加入ret，别忘了加空格

4、最后要把多余的空格去除ret.pop\_back()

**代码：**

class Solution {

public:

string reverseWords(string s) {

int n = s.size(), l, r = n - 1;

string ret;

while(r >= 0){

while(r >= 0 && s[r] == ' ') --r; // clear spaces

if(r < 0) break;

for(l = r; l >= 0 && s[l] != ' '; --l); // fetch word

ret += (s.substr(l + 1, r - l) + " ");

r = l;

}

if(ret.size()) ret.pop\_back();

return ret;

}

};