

按单词下标区间翻转文章内容

题目描述：

给定一段英文文章片段，由若干单词组成，单词间以空格间隔，单词下标从 0 开始。

请翻转片段中指定区间的单词顺序并返回翻转后的内容。

例如给定的英文文章片段为 *"I am a developer"*，翻转区间为 *[0,3]*，则输出 *"developer a am I"*。

String reverseWords(String s, int start, int end)

输入描述：

使用换行隔开三个参数，第一个参数为英文文章内容即英文字符串，第二个参数为待翻转内容起始单词下标，第三个参数为待翻转内容最后一个单词下标。

输出描述：

翻转后的英文文章片段所有单词之间以一个半角空格分隔进行输出

补充说明：

英文文章内容首尾无空格

示例 1

输入：

I am a developer

1

2

输出：

I a am developer

说明：

示例 2

输入:

```
hello world
```

```
-1
```

```
1
```

输出:

```
world hello
```

说明:

下标小于 0 时, 从第一个单词开始

示例 3

输入:

```
I am a developer
```

```
0
```

```
5
```

输出:

```
developer a am I
```

说明:

下标大于实际单词个数, 则按最大下标算

示例 4

输入:

```
I am a developer
```

```
-2
```

-1

输出:

I am a developer

说明:

翻转区间无效时，不做翻转

```
#include <iostream>
```

```
#include <vector>
```

```
#include <algorithm>
```

```
using namespace std;
```

```
int main() {
```

```
    vector<string> strs;
```

```
    string temp;
```

```
    while (cin >> temp) {
```

```
        strs.push_back(temp);
```

```
    }
```

```
    int n = strs.size();
```

```
    int left = stoi(strs[n-2]);
```

```
    int right = stoi(strs[n-1]);
```

```
    if (left < 0) left = 0;
```

```
    if (right >= n - 2) right = n - 3;
```

```
    if (left < right) reverse(&strs[left], &strs[right+1]);
```

```
    for (int i = 0; i < n - 2; i++) {
```

```
        cout << strs[i] << " ";
```

```
}
```

```
}
```

```
// 64 位输出请用 printf("%lld")
```