# 题目

给定两个字符串, A和B。

A的旋转操作就是将A最左边的字符移动到最右边。 例如, 若A= 'abcde'，在移动一次之后结果就是'bcdea' 。如果在若干次旋转操作之后，A能变成B，那么返回True。

示例 1:

输入: A = 'abcde', B = 'cdeab'

输出: true

示例 2:

输入: A = 'abcde', B = 'abced'

输出: false

注意：

A和B长度不超过100。

# 分析

## 方法一：加倍字符串

class Solution {

public:

    bool rotateString(string s, string goal) {

        if(s.length()!=goal.length())

            return false;

        string tmpStr = s+s;

        if(tmpStr.find(goal)!=string::npos)

            return true;

        return false;

    }

};

## 方法二：栈

**思路：**

一般匹配类的问题都可以采用栈解决。

**代码：**