

## 459. Repeated Substring Pattern

Given a non-empty string check if it can be constructed by taking a substring of it and appending multiple copies of the substring together. You may assume the given string consists of lowercase English letters only and its length will not exceed 10000.

### Example 1:

**Input:** "abab"  
**Output:** True  
**Explanation:** It's the substring "ab" twice.

### Example 2:

**Input:** "aba"  
**Output:** False

### Example 3:

**Input:** "abcabcabcabc"  
**Output:** True  
**Explanation:** It's the substring "abc" four times. (And the substring "abcabc" twice.)

### Intution:

Following is the cleverest way to do solve this question

let  $s = abbaabbaabba$

we can see that  $s$  can be formed by repeating  $abba$

now approach is

1. form a new string  $newS = s + s$ , here,  $newS = abbaabbaabbaabbaabbaabba$
2. remove first and last character of  $newS = bbaabbaabbaabbaabbaabb$
3.  $s$  can be formed by repeating a substring if  $s$  is also present in  $newS$ ,  $bbaabbaabbaabbaabbaabb ==> return true$

### **CODE:**

```
bool repeatedSubstringPattern(string s) {  
    return (s+s).substr(1,2*(s.length()) - 2).find(s) != -1;  
}
```