

# AcWing-3325： Kick\_Start

## 题目描述

Ksenia 非常喜欢读书，因此每天她都会从自己最喜欢的书中选取一段内容进行阅读，然后再开始她早晨的其他活动。

一段内容可以看作是整个文本中的一个子字符串。

Ksenia 有点迷信，她坚信如果阅读的这段内容是以字符串 **KICK** 开头，然后中间包含 0 个或多个字符，最后以字符串 **START** 结尾，即使这段内容没什么意义，她的一天也会非常的幸运。

给定这本书的全部文本内容，请你数一数在这本书变得破旧不堪，Ksenia 不得不再买新书之前，共有多少个幸运片段可供她阅读。

只要两个片段的起始位置或结束位置不同，就认为这两个片段是不同的，即使它们包含的内容完全相同。

还需注意，不同片段之间可能会有重叠部分。

**输入格式**

第一行包含整数  $T$ ，表示共有  $T$  组测试数据。

每组数据共一行，包含一个仅由大写字母构成的字符串  $S$ 。

**输出格式**

每组数据输出一个结果，每个结果占一行。

结果表示为 **Case #x: y**，其中  $x$  为组别编号（从 1 开始）， $y$  为幸运片段的数量。

**数据范围**

$1 \leq T \leq 100,$   
 $1 \leq |S| \leq 10^5$

## 输入样例：

```
3
AKICKSTARTPROBLEMNAME DKICKSTART
STARTUNLUCKYKICK
KICKXKICKXSTARTXKICKXSTART
```

## 输出样例：

```
Case #1: 3
Case #2: 0
Case #3: 5
```

难度：	简单
时/空限制：	3s / 64MB
总通过数：	904
总尝试数：	1479
来源：	Google Kickstart2020 Round G Problem A
算法标签	▼

## 算法求解

枚举问题，题目本意是要找出所有 KICK 开头 START 结尾的字符串个数，可以通过一次遍历字符串实现：

使用一个变量 `nums` 记录当前的 KICK 个数

- 遇到一个 KICK，`nums` 自增
- 遇到一个 START，结果增加一个 `nums` 值，即所有在他前面的 KICK 都可以与之组成一个符合要求的字符串。

可以通过 `substr` 函数实现。

Code:

```
//
//  main.cpp
//  3325-kick_start
//
//  Created by MacBook Pro on 2023/8/11.
//

#include <iostream>
using namespace std;

int main() {
    int T;
    scanf("%d",&T);
    for(int cases=1;cases<=T;cases++){
        string s;
        cin>>s;
        int nums = 0;
        int ans = 0;
        for(int i=0;i<s.size();i++){
            if(s.substr(i,4)=="KICK"){
                nums++;
                i+=2;
            }
            else if(s.substr(i,5)=="START"){
                ans += nums;
                i+=4;
            }
        }
        printf("Case #%d: %d\n",cases,ans);
    }
}
```