

字符串划分

题目描述：

给定一个小写字母组成的字符串 s ，请找出字符串中两个不同位置的字符作为分割点，使得字符串分成的三个连续子串且子串权重相等，注意子串不包含分割点。

若能找到满足条件的两个分割点，请输出这两个分割点在字符串中的位置下标，若不能找到满足条件的分割点请返回 $0,0$ 。

子串权重计算方式为：子串所有字符的 **ASCII** 码数值之和。

输入描述：

输入为一个字符串，字符串由 $a \sim z$ ，26 个小写字符组成， $5 \leq \text{字符串长度} \leq 200$ 。

输出描述：

输出为两个分割点在字符串中的位置下标，以逗号分隔

补充说明：

只考虑唯一解，不存在一个输入多种输出解的情况

示例 1

输入：

acdbbbca

输出：

2, 5

说明：

以位置 2 和 5 作为分割点，将字符串分割为 ac ， bb ， ca 三个子串，每一个的子串权重都为 196，输出为：2,5

示例 2

输入：

abcabc

输出：

0, 0

说明：

找不到符合条件的分割点，输出为 0,0

```
#include <iostream>
```

```
#include <vector>
```

```
#include <string>
```

```

using namespace std;

int main()
{
    string strInput;
    cin >> strInput;
    /*if(strInput.size() == 2)
    {
        cout << "0,1" << endl;
        return 0;
    }
    if(strInput.size() < 5)
    {
        cout << "0,0" << endl;
        return 0;
    }*/
    int start = 1;
    int end = strInput.size() - 2;
    int num1 = strInput[0]; // 第一个字符串值
    int num2 = 0; // 第二个字符串值
    int num3 = strInput[strInput.size() - 1]; // 第三个字符串值
    for(int i = 2; i < strInput.size() - 2; ++i)
    {
        num2 += strInput[i];
    }
    bool isHas = false;
    while((end - start) >= 2)
    {
        if(num1 == num3)
        {
            if(num1 == num2)
            {
                isHas = true;
                break;
            }
            else
            {
                num3 += strInput[end];
                end--;
                num2 -= strInput[end];
                num1 += strInput[start];
                start++;
                num2 -= strInput[start];
            }
        }
    }
}

```

```

        }
    }
    else if(num1 > num3)
    {
        num3 += strInput[end];
        end--;
        num2 -= strInput[end];
    }
    else
    {
        num1 += strInput[start];
        start++;
        num2 -= strInput[start];
    }
}
if(isHas)
{
    cout << start << "," << end << endl;
}
else
{
    cout << "0,0" << endl;
}
return 0;
}

```