

题目描述：

给定非空字符串  $s$ ，将该字符串分割成一些子串，使每个子串的 ASCII 码值的和均为水仙花数。

1、若分割不成功，则返回 0

2、若分割成功且分割结果不唯一，则返回 -1

3、若分割成功且分割结果唯一，则返回分割后子串的数目

输入描述：

1、输入字符串的最大长度为 200

输出描述：

根据题目描述中情况，返回相应的结果

补充说明：

“水仙花数”是指一个三位数，每位上数字的立方和等于该数字本身，如 371 是“水仙花数”，因为： $371 = 3^3 + 7^3 + 1^3$

示例 1

输入：

abc

输出：

0

说明：

分割不成功

示例 2

输入：

f3@d5a8

输出：

-1

说明：

分割成功但分割结果不唯一，可以分割为两组，一组 "f3" 和 "@d5a8"，另外一组 "f3@d5" 和 "a8"

示例 3

输入：

AXdddF

输出：

2

说明：

成功分割且分割结果唯一，可以分割 “AX”(153) 和 "dddF"(370) 成两个子串

import sys

```
s = sys.stdin.readline().strip()
```

```
def isFlowerNum(num):
```

```
    sums = 0
```

```
    n = num
```

```
    while n > 0:
```

```
        sums += (n%10)**3
```

```
        n = n//10
```

```

    if num == sums:
        return True
    else:
        return False

def getStrAsciiSum(s):
    sums = 0
    for i in s:
        sums += ord(i)
    return sums

def spStr(s):
    ways = 0
    groups = 0
    for i in range(1, len(s)):
        if isFlowerNum(getStrAsciiSum(s[0:i])):
            #print(s[:i], s[i:])
            ans = spStr(s[i:])
            if ans[0] == -1:
                return -1, 0
            elif ans[1] != 0:
                ways += ans[0]
                groups += ans[1] + 1
    if isFlowerNum(getStrAsciiSum(s)):
        #print(s)
        groups += 1
        ways += 1

    return ways, groups

a = spStr(s)
if a[0] > 1:
    print(-1)
else:
    print(a[1])
#print(spStr(s))

```