

题目描述：

有 N 个正整数组成的一个序列。给定整数 sum ，求长度最长的连续子序列，使他们的和等于 sum ，返回此子序列的长度，如果没有满足要求的序列，返回-1。

输入描述：

序列：1,2,3,4,2

sum：6

输出描述：

序列长度：3

补充说明：

输入序列仅由数字和英文逗号构成，数字之间采用英文逗号分隔；

序列长度： $1 \leq N \leq 200$ ；

输入序列不考虑异常情况，由题目保证输入序列满足要求。

示例 1

输入：

1,2,3,4,2

6

输出：

3

说明：

解释：1,2,3 和 4,2 两个序列均能满足要求，所以最长的连续序列为 1,2,3，因此结果为 3

示例 2

输入：

1,2,3,4,2

20

输出：

-1

说明：

解释：没有满足要求的子序列，返回-1

#

import functools

import sys

from collections import Counter, defaultdict

import copy

from itertools import permutations

import re

import math

import sys

#

m = int(input())

while m > 0:

```

#     m -= 1
#     n = int(input())
#     a = []
#     for i in range(n):
#         x, y = map(int, input().split())
#         a.append([y, x])
#     a.sort(reverse=True)
#     ans = 0
#     now = 0
#     for y, x in a:
#         now += x
#         ans = max(ans, now + y)
#     print(ans)

```

#coding=utf-8

本题为考试多行输入输出规范示例，无需提交，不计分。

import sys

if __name__ == "__main__":

读取第一行的 n

n = int(sys.stdin.readline().strip())

ans = 0

for i in range(n):

读取每一行

line = sys.stdin.readline().strip()

把每一行的数字分隔后转化成 int 列表

values = list(map(int, line.split()))

for v in values:

ans += v

print(ans)

第一题 95%版本

遇到 0 1 先不移

ptin=list(map(int,input().split(',')))

target=int(input())

l,r=0,0

n=len(ptin)

temp=ptin[0]

maxlen=-1

while l<n:

temp=temp+ptin[r]

if temp==target:

templen=r-l+1

maxlen=max(maxlen,templen)

```
        if r+1<n:
            r+=1
            temp = temp + ptin[r]
        # temp -= ptin[l]
        # l += 1
    else:
        break

elif temp>target:
    temp-=ptin[l]
    l+=1
else:# temp<target
    if r+1<n:
        r+=1
        temp = temp + ptin[r]
    else:
        break

print(maxlen)
```