

找

终

点

题目描述：给定一个正整数数组，设为nums，最大为100个成员，求从第一个成员开始，正好走到数组最后一个成员，所使用的最少步骤数。

要求：

- 1、第一步必须从第一元素开始，且 $1 \leq \text{第一步的步长} < \text{len}/2$ （len为数组的长度，需要自行解析）。
- 2、从第二步开始，只能以所在成员的数字走相应的步数，不能多也不能少，如果目标不可达返回-1，只输出最少的步骤数量。
- 3、只能向数组的尾部走，不能往回走。

输入描述：由正整数组成的数组，以空格分隔，数组长度小于100，请自行解析数据数量。

输出描述：正整数，表示最少的步数，如果不存在输出-1

补充说明：

示例

示例1

输入：7 5 9 4 2 6 8 3 5 4 3 9

输出：2

说明：第一步：第一个可选步长选择2，从第一个成员7开始走2步，到达9；第二步：从9开始，经过自身数字9对应的9个成员到最后。

示例2

输入：1 2 3 7 1 5 9 3 2 1

输出：-1

说明：

```
1  import sys
2
3  arr=[]
4  tgt=[]
5
6  for line in sys.stdin:
7      a = line.split()
8      arr=list(map(lambda x:int(x),a))
9  l=len(arr)
10 tgt=[9999]*l
11 tgt[-1]=0
12 for i in range(l):
13     loc=l-1-i
14     a=arr[loc]
15     if a+loc<l and tgt[a+loc]!=9999:
16         tgt[loc]=tgt[a+loc]+1
17
18 ans=1+min(tgt[:l//2])
19 if ans==10000:
20     print(-1)
21 else:
22     print(ans)
```