```
题目描述:输入一串方波信号,求取最长的完全连续交替方波信号,并将其输出,如果有相同长度的交替方波信号,输出任一即可,方波信号高位用1标识,低
        位用0标识, 如图:
        连续方波信号。
        二进制表示
        说明:
        1) 一个完整的信号—定以0开始然后以0结尾,即010是—个完整信号,但101,1010,0101不是
        2) 输入的一串方波信号是由一个或多个完整信号组成
       3) 两个相邻信号之间可能有0个或多个低位,如0110010,011000010
        4) 同一个信号中可以有连续的高位,如01110101011110001010,前14位是一个具有连续高位的信号
       5) 完全连续交替方波是指10交替,如01010是完全连续交替方波,0110不是
  输入描述:輸入信号字符串(长度>=3且<=1024):
        0010101010110000101000010
       注:输入总是合法的,不用考虑异常情况
 输出描述:输出最长的完全连续交替方波信号串:
       01010
       若不存在完全连续交替方波信号串,输出-1
 补充说明: 输入信号串中有三个信号: 0 01010101010(第一个信号段) 00 01010(第二个信号段) 010(第三个信号段)
       第一个信号虽然有交替的方波信号段,但出现了11部分的连续高位,不算完全连续交替方波,在剩下的连续方波信号串中01010最长
示例 1
输入:
00101010101100001010010
输出:
01010
说明:
def fun(s):
    # 分割波 记录波开始结尾坐标
    n=len(s)
    index=[]
    for i in range(n):
        if i==0:
             if s[i+1]=='1':
                 start=0
        elif i==n-1:
             if s[i-1]=='1':
                 end=i
                 index.append([start,end])
        else:
             if s[i]=='0' and s[i+1]=='1' and s[i-1]=='0':# 001
                 start=i
             if s[i]=='0' and s[i+1]=='0' and s[i-1]=='1':
                 end=i
                 index.append([start, end])
    # 和开始结尾坐标奇偶性相同的必须是 0,不同的必须是 1
    maxlen=-1
```

k=-1

for j in range(len(index)):

```
start,end=index[j]
          flag=True
          for i in range(start,end+1,2):
               if s[i]!='0':
                   flag=False
                    break
          if flag:
               for i in range(start+1,end+1,2):
                   if s[i]!='1':
                         flag=False
                         break
          if flag:
               lentemp=end-start+1
               if lentemp>maxlen:
                    maxlen=lentemp
                    k=j
     if maxlen>0:
         return s[index[k][0]:index[k][1]+1]
     else:
          return -1
s=input()
# s='00101010101100001010010'
print(fun(s))
```