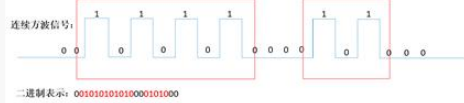


## C-最长方连续方波信号

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:



说明:

- 1) 一个完整的信号一定以0开始然后以0结尾, 即010是一个完整信号, 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011101010111100001010, 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: 输入信号字符串 (长度 $\geq 3$ 且 $\leq 1024$ ):  
001010101010110000101000010  
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 010101010110(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: 0010101010101100001010010  
输出: 01010  
说明:

```
#include <stdio.h>
#include <string.h>
int main() {
    int c;
    int max = 0;
    char data[1025];
    scanf("%s", data);
    c = strlen(data);
    int i = 0;
    int j = 0;
    int flag = 0;
    int h = 0;
    int num = 0;
    int search(char data[], int i, int flag, int h);
    for(j=0; j<c; j++)
    {
        num=search(data, j, flag, h);
        if(num>=3&&num>max)
        {
            max = num;
        }
        flag = 0;
    }
    if(max == 0)
```

```

{
    printf("-1");
    return 0;
}
else {
    for(j=0;j<max;j++)
    {
        if(j%2 == 0)
        {
            printf("0");
        }
        else {
            printf("1");
        }
    }
}
return 0;
}
int search(char data[],int i,int flag,int h)
{
    if(flag == 0)
    {
        if(data[i] == '0')
        {
            i++;
            h++;
            flag = 1;
            h=search(data, i,flag,h);
        }
        else {
            h = 0;
            return h;
        }
    }
    else {
        if(data[i] == '1')
        {
            i++;
            h++;
            flag = 0;
            h=search(data, i,flag,h);
        }
        else {
            return h;
        }
    }
}

```

```
    }  
  }  
  return h;  
}
```