

找车位

题目描述：停车场有一横停车位，0 代表没有停车，1 代表有车。至少停了一辆车在车位上，也至少有一个空位没有停车。 为了防剐蹭，需为停车人找到一个车位，使得距停车人的车最近的车辆的距离是最大的，返回此时的最大距离。 输入描述：1、一个用半角逗号分割的停车标识字符串，停车标识为 0 或 1，0 为空位，1 为已停车。 2、停车位最多 100 个。

输出描述：输出一个整数记录最大距离。

示例1

输入：1,0,0,0,0,1,0,0,1,0,1

输出：2

说明：当车停在第3个位置上时，离其最近的的车距离为2（1到3）。

当车停在第4个位置上时，离其最近的的车距离为2（4到6）。

其他位置距离为1。

因此最大距离为2。

```

1  const rl = require("readline").createInterface({ input: process.stdin });
2  var iter = rl[Symbol.asyncIterator]();
3  const readline = async () => (await iter.next()).value;
4
5  void async function () {
6    while(line=await readline()){
7      //就是找出最长的连续0序列
8      let a=line.split(',');
9      //n是字符串长度,拆成数组
10     let n=a.length;
11     let ans=0;
12     //遍历数组,找有车的位置
13     for(let i=0;i<n;i++){
14       if(a[i]== '1') continue;
15       let mi=n;
16       for(let j=i-1;j>=0;j--){
17         if(a[j]== '1'){
18           mi=Math.min(mi,i-j);
19           break;
20         }
21       }
22       for(let j=i+1;j<n;j++){
23         if(a[j]== '1'){
24           mi=Math.min(mi,j-i);
25         }
26       }
27       ans=Math.max(ans,mi);
28     }
29     console.log(ans);
30   }
31   // Write your code here
32   while(line = await readline()){
33     let tokens = line.split(' ');
34     let a = parseInt(tokens[0]);
35     let b = parseInt(tokens[1]);
36     console.log(a + b);
37   }
38 }

```