

一种新型疾病，开始在全世界传播。尽可能多的保持“社交距离”显得特别的重要，这样可以预防感染。

现在有一个狭长的宾馆，有一排共 N 个房间 ($2 \leq N \leq 10^5$)。有些房间里目前有客人，有些目前空着。得知“社交距离”的重要性，宾馆的管理者希望使得 D 尽可能大，其中 D 为最近的两个有客人的房间的距离。例如，如果房间 3 和 8 是最近的有客人的房间，那么 $D=5$ 。

最近两个客人新来到这个狭长的宾馆，宾馆的管理者他需要决定将他们分配到哪两个之前空着的房间。请求出宾馆的管理者如何安排这两位新来的客人，使得 D 仍然尽可能大。宾馆的管理者不能移动任何已有的客人；他只想要给新来的客人分配房间。

输入格式（文件名：socdist1.in）：

输入的第一行包含 N 。下一行包含一个长为 N 的字符串，由 0 和 1 组成，描述宾馆里的房间。0 表示空着的房间，1 表示有客人的房间。字符串中包含至少两个 0，所以有至少有足够的房间安置两位新来的客人。

输出格式（文件名：socdist1.out）：

输出 以最优方案在加入两位新来的客人后，可以达到的最大 D 值。

输入样例：

```
14
10001001000010
```

输出样例：

```
2
```

在这个例子中，管理者可以以这样的方式加入客人，使得房间分配变为 $10x010010x0010$ ，其中 x 表示新来的客人。此时 $D=2$ 。不可能在加入客人之后取到更大的 D 值。

测试点性质：

- 测试点 2-6 满足 $N \leq 10$ 。
- 测试点 7-8 满足 $N \leq 100$ 。
- 测试点 9-11 满足 $N \leq 5000$ 。
- 测试点 12-15 没有额外限制。