Jumping on the Clouds

by Shafaet

Problem | Submissions | Leaderboard | Discussions | Editorial

Emma is playing a new mobile game involving n clouds numbered from 0 to $n-1$. A player initially starts out on cloud c_0, and they must jump to cloud c_{n-1}. In each step, she can jump from any cloud i to cloud $i+1$ or cloud $i+2$.

There are two types of clouds, *ordinary clouds* and *thunderclouds*. The game ends if Emma jumps onto a thundercloud, but if she reaches the last cloud (i.e., c_{n-1}), she wins the game!

Can you find the minimum number of jumps Emma must make to win the game? It is guaranteed that clouds c_0 and c_{n-1} are ordinary-clouds and it is *always possible* to win the game.

Input Format

The first line contains an integer, n (the total number of clouds).
The second line contains n space-separated binary integers describing clouds $c_0, c_1, \ldots, c_{n-1}$.

- If $c_i = 0$, the i^{th} cloud is an ordinary cloud.
- If $c_i = 1$, the i^{th} cloud is a thundercloud.

Constraints

- $2 \leq n \leq 100$
- $c_i \in \{0, 1\}$
- $c_0 = c_{n-1} = 0$

Output Format

Print the minimum number of jumps needed to win the game.

Sample Input 0

```
7
0 0 1 0 0 1 0
```

Sample Output 0

```
4
```

Sample Input 1

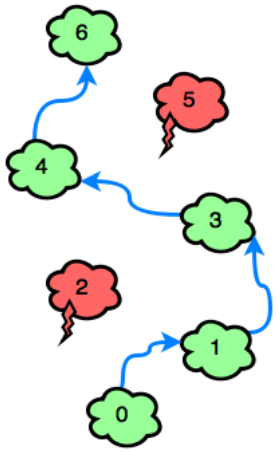
```
6
0 0 0 0 1 0
```

Sample Output 1

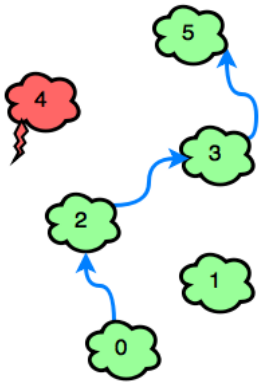
```
3
```

Explanation

Sample Case 0:
Because c_2 and c_5 in our input are both 1, Emma must avoid c_2 and c_5. Bearing this in mind, she can win the game with a minimum of 4 jumps:



Sample Case 1:
The only thundercloud to avoid is c_4. Emma can win the game in 3 jumps:



Submissions: 19846
Max Score: 20
Difficulty: Easy

Rate This Challenge:

More

Current Buffer (saved locally, editable) C#

```
using System;
using System.Collections.Generic;
using System.IO;
using System.Linq;
class Solution {

    static void Main(String[] args) {

        int n = int.Parse(Console.ReadLine());
            int[] clouds = Array.ConvertAll(Console.ReadLine() .Split(' '), e => int.Parse(e));


            int jumps = 0;
            for (int i = 0; i + 1 < clouds.Length; )
            {
                if (clouds[i + 1] == 1 || (i+2 < clouds.Length && clouds[i+2] ==0))
                {
                    jumps++;
                    i += 2;
                }
                else
                {
                    jumps++;
                    i++;
                }
            }
            Console.WriteLine(jumps);


    }
}

```

Line: 27 Col: 38

Upload Code as File Test against custom input Run Code Submit Code

Congrats, you solved this challenge!

- ✔ Test Case #0
- ✔ Test Case #1
- ✔ Test Case #2
- ✔ Test Case #3
- ✔ Test Case #4
- ✔ Test Case #5
- ✔ Test Case #6
- ✔ Test Case #7
- ✔ Test Case #8

Next Challenge