Problem: Beautiful Binary String

Alice has a [binary string](https://www.ics.uci.edu/~alspaugh/cls/shr/binaryString.html), , of length . She thinks a binary string is beautiful if and only if it doesn't contain the [substring](https://en.wikipedia.org/wiki/Substring) .

In one step, Alice can change a  to a  (or vice-versa). Count and print the minimum number of steps needed to make Alice see the string as beautiful.

**Input Format**

The first line contains an integer,  (the length of binary string ).   
The second line contains a single binary string, , of length .

**Constraints**

* Each character in .

**Output Format**

Print the minimum number of steps needed to make the string beautiful.

**Sample Input 0**

7

0101010

**Sample Output 0**

2

**Sample Input 1**

5

01100

**Sample Output 1**

0

**Sample Input 2**

10

0100101010

**Sample Output 2**

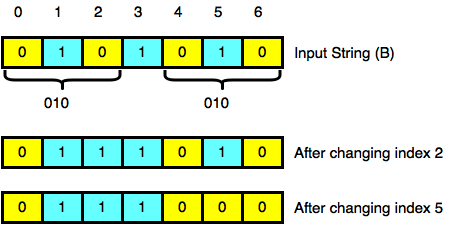
3

**Explanation**

**Sample Case 0:**

In this sample,

The figure below shows a way to get rid of each instance of :



Because we were able to make the string beautiful by changing  characters ( and ), we print .

**Sample Case 1:**

In this sample

The substring  does not occur in , so the string is already beautiful and we print .

Solution

int main() {

int length; string str;

cin>>length; cin>>str;

int count=0;

int a, b, c;

for(int i=0; i<length-2; i++)

{

a=i; b=a+1; c=a+2;

if(str[a]=='0' && str[b]=='1' && str[c]=='0')

{ str[c]='1'; count+=1; i-=1;}

}

cout<<count;

return 0;

}

Elegant Solution

int beautifulBinaryString(String b)

{

return ( ( b.length() - b.replaceAll("010","").length() )/3 );

}

`’Anshul AgGarwal