'''

Ram and Sita are subscribed to a travel service for their respective

business commitments in Rome.

You are provided with four subscription strings, startOfRam,

endOfRam, startOfSita, and endOfSita.

Ram's subscription allows him to be in the city from the dates

startOfRam to endOfRam(inclusive), while Sita's subscription

covers the period from startOfSita to endOfSita (inclusive).

Each subscription date is represented as a 5-character string in

the format "MM-DD", indicating the month and day of the date.

Your task is to determine the total number of days when

they can enjoy their Rome experience together.

Assume that all subscription dates fall within the same calendar year,

which is not a leap year. Note that the number of days per month

can be represented as: [31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31].

Input Format:

-------------

4 space separated strings, startOfRam, endOfRam, startOfSita, and endOfSita.

Output Format:

-------------

An integer result.

Sample Input-1:

----------------

08-15 08-18 08-16 08-19

Sample Output-1:

----------------

3

Explanation:

------------

Ram will be in Rome from August 15 to August 18.

Sita will be in Rome from August 16 to August 19.

They both in Rome together on August 16th, 17th, and 18th,

so the answer is 3.

Sample Input-2:

----------------

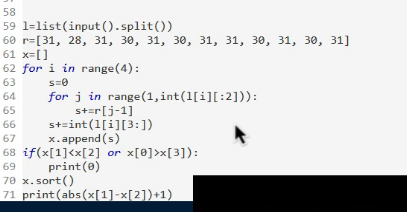
10-01 10-31 11-01 12-31

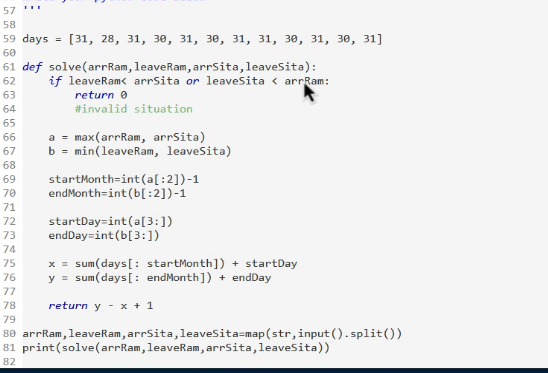
Sample Output-2:

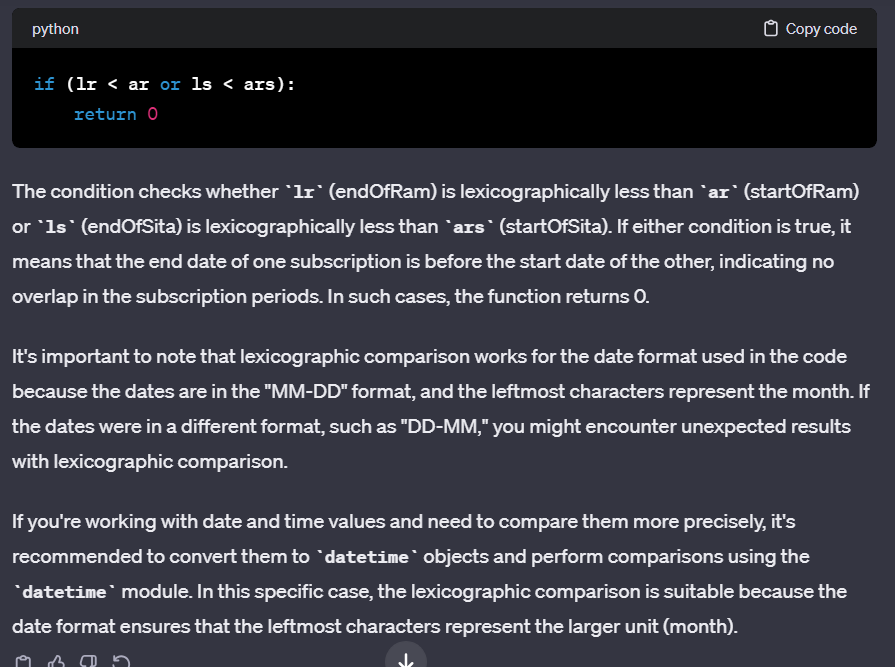
----------------

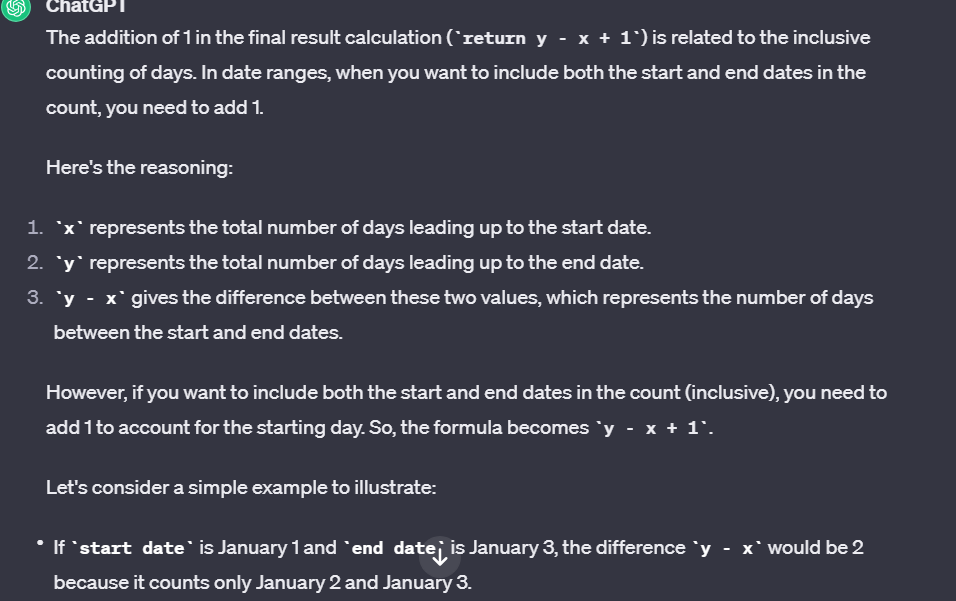
0

Write your python code below









Sayyad and Noel work in the same office but in different departments.

Sayyad needs to share a password protected file with Noel.

Sayyad encrypted the password message and sent to Noel.

The Password is a single word w[] of length 'N' which consists only 2 types of

characters 'A' and 'D'.

Now Noel has to decrypt the message in terms of an integer array ar[]

with the following rules.

-w[i]='A' if ar[i]<ar[i+1]

-w[i]='D' if ar[i]>ar[i+1]

-He has to use the numbers from 0 to N only.

Return the decrypted array which consists of N+1 elements (0 to N).

Constraints:

1 <= N <= 10^5

w[i] is either 'A' or 'D' .

Input Format:

-------------

Line-1: A string represents a word.

Output Format:

--------------

Print an integer array.

Sample Input-1:

---------------

ADAD

Sample Output-1:

----------------

0 4 1 3 2

Explanation:

------------

0 to 4 --> Ascending(A)

4 to 1 --> Descending(D)

1 to 3 --> Ascending(A)

3 to 2 --> Descending(D)

So the sequence is ADAD.

Sample Input-2:

---------------

AAAA

Sample Output-2:

----------------

0 1 2 3 4

Explanation:

------------

0 to 1 --> Ascending(A)

1 to 2 --> Ascending(A)

2 to 3 --> Ascending(A)

3 to 4 --> Ascending(A)

So the sequence is AAAA.

import java.util.\*;

public class Main{

public static void main(String[] args){

Scanner sc=new Scanner(System.in);

String s=sc.next();

ArrayList<Integer>k=new ArrayList<>();

ArrayList<Integer> ans=new ArrayList<>();

for(int i=0;i<=s.length();i++){

k.add(i);

}

for (int i=0;i<s.length();i++){

if(s.charAt(i)=='A'){

ans.add(k.get(0));

k.remove(0);

}

else if(s.charAt(i)=='D'){

ans.add(k.get(k.size()-1));

k.remove(k.size()-1);

}

}

ans.add(k.get(0));

System.out.println(ans);

}

}

There is a series of bulbs in which some bulbs are turned on(indicated by 1)

and others are off(indicated by 0).

A contiguous sub-series of bulbs CSB with following rules.

1. Number of bulbs with status as 'on' and 'off' are same.

2. Bulbs which are 'on' and 'off' are grouped together.

You will be given the series as a string of 1's and 0's.

Your task is to find the count of all the CSBs exist in the given series.

Input Format:

-------------

A string, indicates bulbs series.

Output Format:

--------------

Print an integer result.

Sample Input-1:

---------------

010011000110

Sample Output-1:

----------------

9

Explanation:

------------

01-3, 10-3, 0011-2, 1100-1

Sample Input-2:

---------------

0101010

Sample Output-2:

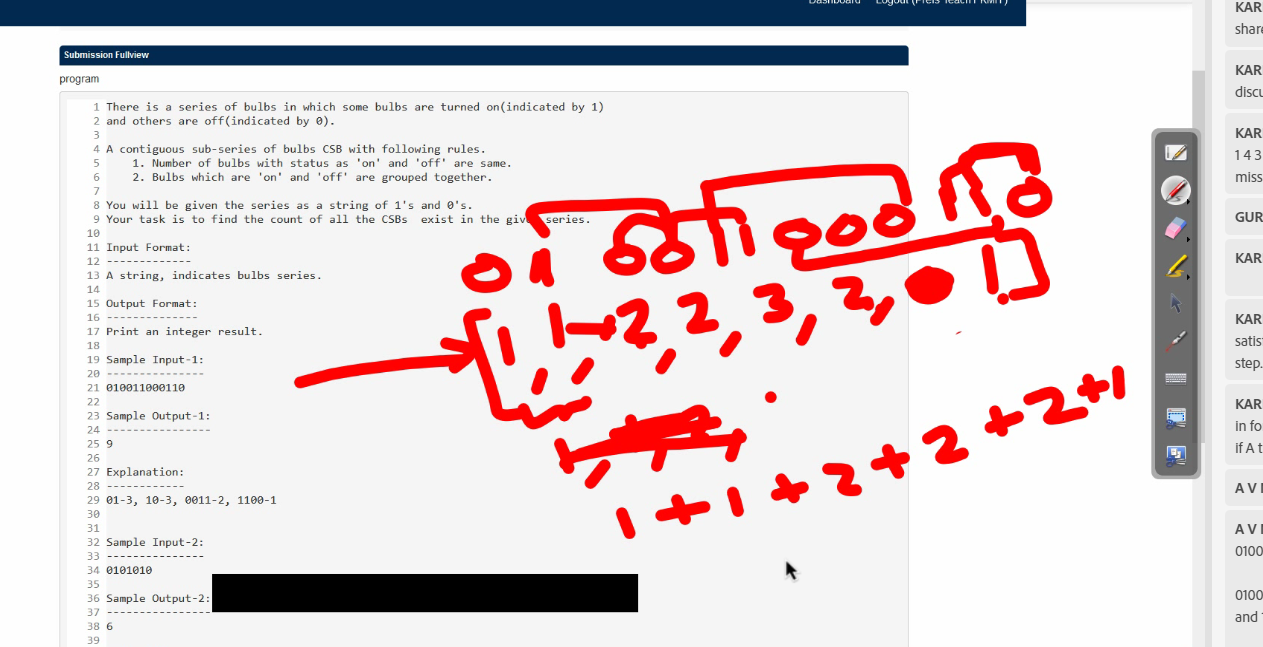
----------------

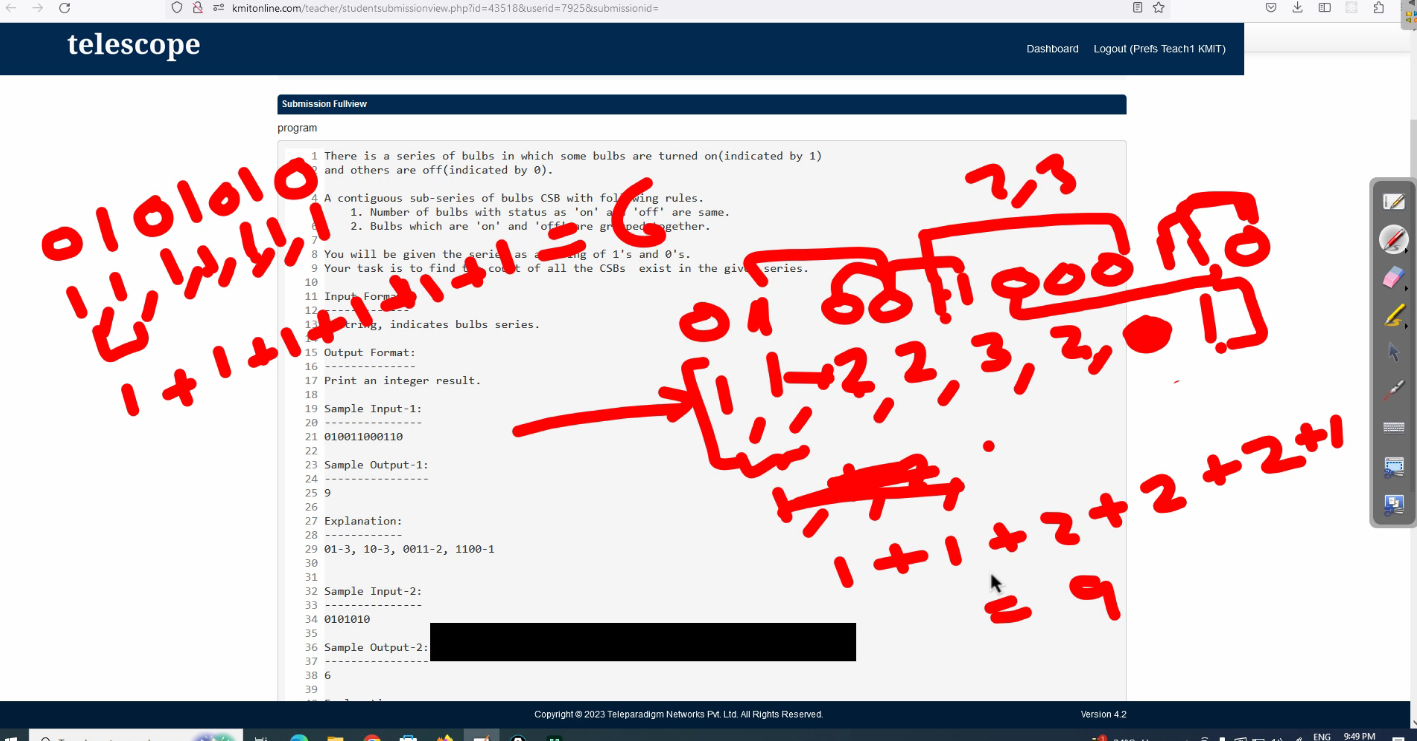
6

Explanation:

------------

01-3, 10-3





import java.util.\*;

public class Main{

public static void main(String[] args){

Scanner sc=new Scanner(System.in);

List<Integer> l=new ArrayList<>();

String s=sc.next();

int i=0;

while(i<s.length()){

int cz=0;

int co=0;

if(s.charAt(i)=='0'){

while( i<s.length() && s.charAt(i)!='1'){

cz+=1;

i+=1;

}

l.add(cz);

}

else if(s.charAt(i)=='1'){

while( i<s.length() && s.charAt(i)!='0'){

co+=1;

i+=1;

}

l.add(co);

}

}

int sum=0;

for(i=0;i<l.size()-1;i++){

sum+=Math.min(l.get(i),l.get(i+1));

}

System.out.println(sum);

}

}