'''

"Emphatic Pronunciation" of a given word is where we take the word and

replicate some of the letter to emphasize their impact.

Instead of saying 'oh my god', someone may say "ohhh myyy goddd",

We define emphatic pronunciation of a word, which is derived by replicating

a group (or single) of letters in the original word.

So that the replicated group is atleast 3 characters or more and

greater than or equal to size of original group.

For example Good -> Goood is an emphatic pronunciation,

but Goodd is not because in Goodd the 'd' are only occuring twice consecutively.

In the question you are given the "Emphatic pronunciation" word,

you have to findout how many words can legal result in the

"emphatic pronunciation" word.

Input Format:

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

Line-1 -> A String contains a single word, Emphatic Pronunciation word

Line-2 -> Space seperated word/s

Output Format:

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

Print an integer as your result

Sample Input-1:

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

Goood

Good

Sample Output-1:

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

1

Sample Input-2:

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

heeelllooo

hello hi helo

Sample Output-2:

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

2

Only 37.5% test cases passed

'''

s=input()

l=list(input().split())

ct=0

d=dict()

for i in range(len(s)):

if s[i] not in d:

d[s[i]]=1

continue

if(i-1>=0 and s[i-1]==s[i]):

d[s[i]]+=1

def dic(s):

di=dict()

for i in range(len(s)):

if s[i] not in di:

di[s[i]]=1

continue

if(i-1>=0 and s[i-1]==s[i]):

di[s[i]]+=1

return di

for i in l:

d1=dic(i)

for j in d1:

if(j in d and d[j]>=3 and d1[j]<=d[j]):

ct+=1

break

print(ct)

Tejaswi playing a game. Game has a display of N numbers in a row, nums[].

Tejaswi has to pick the numbers one after the other.

Once Tejaswi picks a number num[i], she will score num[i] points and

the number will be disappared on the board, simultaneously all the numbers

having the value num[i]+1 or num[i]-1 also disappears.

Tejaswi has to select the numbers, such that no more numbers will be left over

on the board.

Tejaswi wants to score maximum number of points.

Initially Tejaswi has 0 points with her.

Can you help Tejaswi to score maximum number of points.

Input Format:

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

Line-1 -> an integers N, number of numbers on the game board.

Line-2 -> N space separated integers, numbers on the game board nums[].

Output Format:

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

Print an integer as result, maximum score she can get.

Sample Input-1:

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

3

4 5 3

Sample Output-1:

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

8

Explanation:

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

Pick a number 5 to score 5 points, simultaneously 4 is disappeared from display.

Then, pick number 3 to score 3 points.

Totally 8 is the score.

Sample Input-2:

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

6

4 4 5 5 5 6

Sample Output-2:

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

15

Explanation:

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

Pick a number 5 to score 5 points, simultaneously all 4's and 6 are disappeared

from display. Then, again pick the number 5 to score 5 points.

And again pick the number 5 to score 5 points. Totally 15 is the score.

Only 25% test cases passed

import java.util.\*;

public class Main{

public static void main(String[] args){

Scanner sc=new Scanner(System.in);

int n=sc.nextInt();

int[] arr=new int[n];

// boolean[]vis=new boolean[n];

for(int i=0;i<n;i++){

arr[i]=sc.nextInt();

// vis[i]=sc.next();

}

HashMap<Integer,Integer> hs=new HashMap<>();

HashMap<Integer,Integer> dup=new HashMap<>();

for (int i : arr){

if(hs.containsKey(i)){

hs.put(i,hs.get(i)+1);

// dup.put(i,hs.get(i)+1);

}

else{

hs.put(i,1);

// dup.put(i,1);

}

}

for (Integer i: hs.keySet()){

dup.put(i,hs.get(i));

}

int max=Integer.MIN\_VALUE;

// System.out.println(hs);

// System.out.println(dup);

for (Integer i : hs.keySet()){

int sum=0;

// for ()

// for(int j=0;j<hs.get(i);j++){

if(dup.containsKey(i-1)){

dup.put(i-1,0);

}

if(dup.containsKey(i+1)){

dup.put(i+1,0);

}

// System.out.println("..."+dup);

// }

for (Integer j : dup.keySet()){

// if(dup.get(j)!=0){

sum+=dup.get(j)\*j;

// }

// dup.get(i)=hs.get(i);

dup.put(j,hs.get(j));

}

// System.out.println(dup);\*9

if(sum>max){

max=sum;

}

// System.out.println(sum);

// for(Integer )

}

System.out.println(max);

}

}

Ananth interested in creating the acronyms for any word. The definition

of acronym is another word with a concatenation of its first letter,

the number of letters between the first and last letter, and its last letter.

If a word has only two characters, then it is an acronym of itself.

Examples:

- Acronym of 'fog' is f1g'.

- Acronym of 'another' is 'a5r'.

- Acronym of 'ab' is 'ab'.

You are given a list of vocabulary, and another list of words.

Your task is to check,each word with the vocabulary and

return "true" if atleast one of the following rules satisfied:

1. acronym of the word should not match with any of the acronyms of vocabulary

2. if acronym of the word matches with any of the acronyms of vocabulary

'the word' and matching words in vocabulary should be same.

Otherwise, return 'false'.

Input Format:

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

Line-1: Space separated strings, vocabulary[]

Line-2: Space separated strings, words[]

Output Format:

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

Print N boolean values, where N = words.length

Sample Input-1:

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

cool bell cool coir move more mike

cool char move

Sample Output-1:

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

true false false

import java.util.\*;

public class Main{

public static void main(String[] args){

Scanner sc=new Scanner(System.in);

String vo=sc.nextLine();

String wo=sc.nextLine();

String[] voc=vo.split(" ");

String[] words=wo.split(" ");

ArrayList<String> vocab=new ArrayList<>();

ArrayList<String> vocabl=new ArrayList<>();

ArrayList<String> wordl=new ArrayList<>();

for (String i : voc){

vocab.add(i);

vocabl.add(acronym(i));

}

for(String i : words){

wordl.add(acronym(i));

}

for (int i=0;i<wordl.size();i++){

if(!vocabl.contains(String.valueOf(wordl.get(i)))){

System.out.print(true+" ");

continue;

}

// for(int j=0;j<vocabl.size();j++){

// if(wordl.get(i)==vocabl.get(j)){

// if

// }

// }

boolean flag=true;

for(int j=0;j<vocab.size();j++){

if((vocabl.get(j)).equals(wordl.get(i))){

if(!(vocab.get(j)).equals(words[i])){

// System.out.print(false+" ");

flag=false;

// continue;

break;

}

}

}

// if(vocab.contains(words[i])){

System.out.print(flag+" ");

// // continue;

// }

// else{

// System.out.print(false+" ");

// }

}

}

public static String acronym(String s){

if(s.length()<1){

return s;

}

if(s.length()==2){

return s.charAt(0)+""+""+s.charAt(1);

}

return s.charAt(0)+""+(s.length()-2)+""+s.charAt(s.length()-1);

}

}