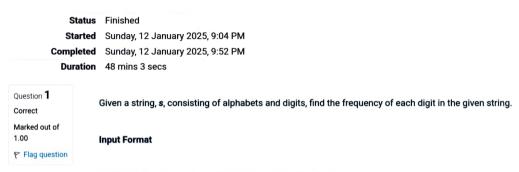
GE23131-Programming Using C-2024





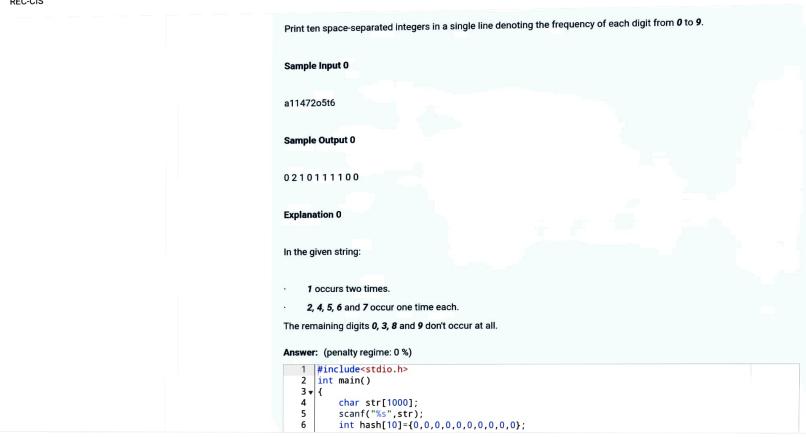
The first line contains a string, *num* which is the given number.

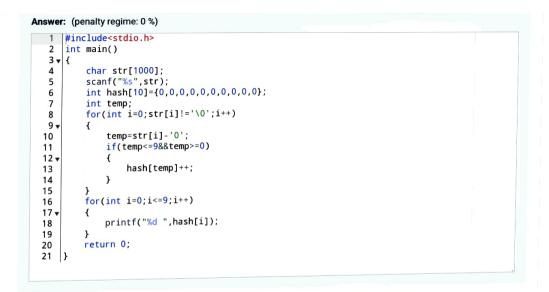
Constraints

1 ≤ len(num) ≤ 1000

All the elements of num are made of English alphabets and digits.

Output Format





	Input	E	хp	ec	te	d						G	ot									
~	a11472o5t6	0	2	1	0	1	1	1	1	0	0	0	2	1	0	1	1	1	1	0	0	~
_	lw4n88j12n1	0	2	1	0	1	0	0	0	2	0	0	2	1	0	1	0	0	0	2	0	~
_	1v888861256338ar0ekk	1	1	1	2	0	1	2	0	5	0	1	1	1	2	0	1	2	0	5	0	~

Marked out of 1.00 Flag question

Question 2

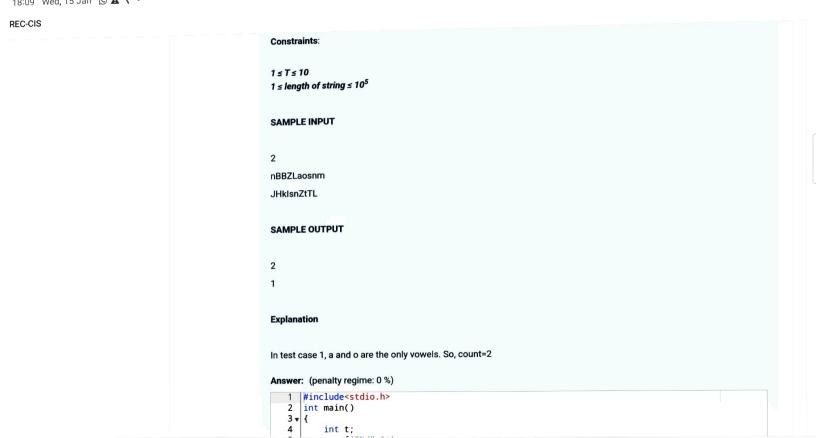
Correct

them. So, he asked you to tell him the count of such trees in the garden. Input: Output:

Note: The following letters are vowels: 'A', 'E', 'I', 'O', 'U', 'a', 'e', 'i', 'o' and 'u'. The first line consists of an integer ${\it T}$ denoting the number of test cases. Each test case consists of only one string, each character of string denoting the alphabet (may be lowercase or uppercase) on a tree in the garden. For each test case, print the count in a new line. Constraints: $1 \le T \le 10$ $1 \le \text{length of string} \le 10^5$ SAMPLE INPUT

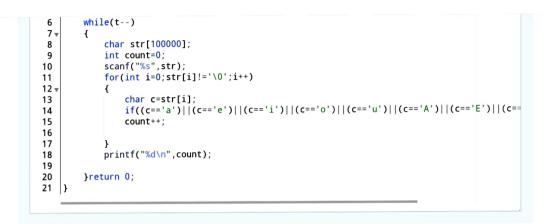
Today, Monk went for a walk in a garden. There are many trees in the garden and each tree has an English alphabet on it.

While Monk was walking, he noticed that all trees with vowels on it are not in good state. He decided to take care of



```
#include<stdio.h>
 2
    int main()
 3 ₹ {
         int t;
        scanf("%d",&t);
        while(t--)
 7 ₩
            char str[100000];
            int count=0;
10
            scanf("%s",str);
11
            for(int i=0;str[i]!='\0';i++)
12 ₩
13
                char c=str[i];
14
                if((c=='a')||(c=='e')||(c=='i')||(c=='o')||(c=='u')||(c=='A')||(c=='E')||(c==
15
                count++;
```

16 17 18 19 20 21	} preture }	rintf("%d	\n", c	ount:
	Input	Expected	Got	
~	2 nBBZLaosnm JHkIsnZtTL	2	2	~
~	2 nBBZLaosnm JHkIsnZtTL	2	2	~





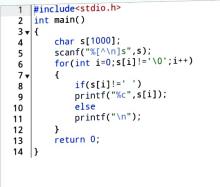
REC-CIS



Given a sentence, s, print each word of the sentence in a new line. **Input Format** The first and only line contains a sentence, s. Constraints 1 ≤ len(s) ≤ 1000 **Output Format** Print each word of the sentence in a new line. Sample Input 0 This is C Sample Output 0 This is

In the given string, there are three words ["This", "is", "C"]. We have to print each of these words in a new line.

Answer: (penalty regime: 0 %)



Ì		Input	Expected	Got	
	~	This is C	This is	This is	~

```
for(int i=0;s[i]!='\0';i++)

for(int i=0;s[i]!='\0';i++)

if(s[i]!=' ')
    printf("%c",s[i]);
    else
    printf("\n");

return 0;
```

	Input	Expected	Got	
~	This is C	This is C	This is C	~
~	Learning C is fun	Learning C is fun	Learning C is fun	~

Passed all tests! 🗸

Question 4 Correct Marked out of 1.00 Flag question In the second line print the string produced by concatenating \boldsymbol{a} and \boldsymbol{b} ($\boldsymbol{a} + \boldsymbol{b}$). In the third line print two strings separated by a space, a' and b' are the same as a and b, respectively, except that their first characters are swapped. Sample Input abcd Sample Output 42 abcdef ebcd af

Input Format

You are given two strings, a and b, separated by a new line. Each string will consist of lower case Latin characters ('a'-'z').

Output Format

ef

In the first line print two space-separated integers, representing the length of \boldsymbol{a} and \boldsymbol{b} respectively.

REC-CIS	In the second line print the string produced by concatenating a and b ($a + b$)	o).
	In the third line print two strings separated by a space, a' and b' . a' and b' are that their first characters are swapped.	e the same as a and b , respectively, except
	Sample Input	
	abcd ef	
	Sample Output	
	42	
	abcdef	
	ebcd af	
	Explanation	
	a = "abcd"	
	b = "ef"	
	a = 4	
	b = 2	
	a + b = "abcdef"	
	a' = "ebcd"	

10:10 11:04, 10:04:1 2 - 1

```
Answer: (penalty regime: 0 %)
      #include<stdio.h>
      int main()
   3 ₹ {
          char str1[10],str2[10],t;
          int i=0; int j=0;
          int count1=0;int count2=0;
          scanf("%s", str1);
          scanf("%s", str2);
          while(str1[i]!='\0')
   9
  10 ▼
  11
              count1++;
  12
              i++;
  13
          while(str2[j]!='\0')
  14
  15 ▼
  16
              count2++;
  17
              j++;
  18
  19
  20
          printf("%d %d\n",count1,count2);
  21
          printf("%s%s\n",str1,str2);
  22
          t=str1[0];
  23
          str1[0]=str2[0];
  24
          str2[0]=t;
          printf("%s %s",str1,str2);
  26
          return 0;
  27 }
```

```
11
            count1++;
12
            i++;
13
14
        while(str2[j]!='\0')
15 ▼
16
            count2++;
17
            j++;
18
19
20
        printf("%d %d\n",count1,count2);
21
        printf("%s%s\n",str1,str2);
22
        t=str1[0];
23
        str1[0]=str2[0];
24
        str2[0]=t;
25
        printf("%s %s",str1,str2);
26
        return 0;
27 }
```

