

CSDC101 Fundamentals of Computing

Laboratory Exercise 8

PROBLEMS

1) Name

Write a program that inputs a first and last name, separated by a space, into a string variable. Use the string functions to output the first and last initial.

INPUT FORMAT

Input begins with an integer N indicating how many names follow. Succeeding lines are pairs of $s1$ and $s2$ strings, where $s1$ is the first name and $s2$ is the last name.

OUTPUT FORMAT

For each input, output the first and last initial in uppercase with corresponding period. Separate the output with newline.

CONSTRAINTS

$1 \leq N \leq 100$

$s1$ and $s2$ strings comprised of lowercase letters

SAMPLE INPUT

```
3
mark manson
daniel pink
aubrey marcus
```

SAMPLE OUTPUT

```
M.M.
D.P.
A.M.
```

2) Cardo's Words

Cardo type a sequence of words in camel case. The words are string of letters having the following characteristics:

- It is a concatenation of one or more words consisting of English letters only.
- The first letter of the words are in uppercase and the rest are in lowercase.

For example, the string `HelloWorldHello` is composed of **three (3) words**.

The challenge is to create a function named **CountWords()** that will string (compose of sequence of words in camel case) *s* and return the number of words found in a sequence of words in camel case. Be able to use the function to create the complete C++ program.

INPUT FORMAT

The input is a string containing sequence of words in `CamelCase`.

OUTPUT FORMAT

The output is the total number of words found in the string.

Sample Run:

```
Enter sequence of words in CamelCase: MerryChristmasHappyNewYear
No. of words found: 5
Another[y/n]? y
Enter sequence of words in CamelCase: CardoAngProbinsiyano
No. of words found: 3
Another[y/n]? n
```

3) Minimum Substring Flag

Given two strings, A and B , determine if a common substring exists between A and B .

INPUT FORMAT

The first line of the input will contain a single integer T , followed by T number of test cases. Each test case consists of two lines of strings, A and B .

Positions given by A and B begin from 1 (e.g. 1 pertains to the first value in the list).

OUTPUT FORMAT

For each test case, display *YES* if there is a common substring. Otherwise, display *NO*.

CONSTRAINTS

$1 \leq T \leq 100$

$1 \leq |A|, |B| \leq 10^9$

SAMPLE INPUT

```
2
hello
helloworld
hi
world
```

OUTPUT FORMAT

```
YES
NO
```

4) Arrow Through My Heart

Cupid found his way through my heart, shot it with his magical arrow, and, just like that... LOVE!
Find the heart in every string and show how it's been shot with Cupid's enchanted arrow.

INPUT FORMAT

Input consists of several strings S , one in each line.

OUTPUT FORMAT

For each test case, display the string with all substrings equal to "heart" replaced with ">>-h-e-a-r-t->".

CONSTRAINTS

S can be spaces, lowercase letters, or a combination of both.

SAMPLE INPUT

```
Heart
this is some test
dealing with a heartbreak heartaches
he is very downhearted
```

SAMPLE OUTPUT

```
>>-h-e-a-r-t->
this is some test
dealing with a >>-h-e-a-r-t->break >>-h-e-a-r-t->aches
he is very down>>-h-e-a-r-t->ed
```

5) Antonio's Message

After making his reputation as a rebel, Antonio is planning for a fight to stop President Ridge from putting him to jail. The President ordered his army to arrest Antonio. Holed up in a secret place, Antonio seeks the help of his allies to avoid being arrested. With this, he needed to send text messages to them. But doing this will be very risky. Thus, Antonio thought of using a transposition cipher to convert his text messages into code. For example, his intended text message: meet me after work behind the office is coded to as:

`maio efrnf etkdf tebti mrehc ewhee`

The coded text message is generated by writing down the intended text message in square or rectangle form. For example, the intended text message is written in a six by five square (as shown below):

Then, the coded text message is generated by reading down the columns from left to right.

m	e	e	t	m	e
a	f	t	e	r	w
o	r	k	b	e	h
i	n	d	t	h	e
o	f	f	i	c	e

The size of the square or rectangle is determined by the length of the text message.

If the length of the message resulted to a perfect square, that number is used as the number of columns. Otherwise, the number of columns is determined by the smallest square that is larger than the number of characters in the intended message.

For example: an intended text message of 5 characters long will create a 3 columns wide. While an intended text message of 4 characters long will use 2 x 2 square.

With this, help Antonio create the coded messages he will send to his allies to avoid being arrested.

INPUT FORMAT

The input contains lowercase text messages where the spaces and punctuations are removed.

OUTPUT FORMAT

The output contains coded text messages.

SAMPLE RUN:

Antonio's Text Message: protectoursovereignty
Coded Text Message: pcsey rtoi oovg tuen errt