Problem H
Hotfix

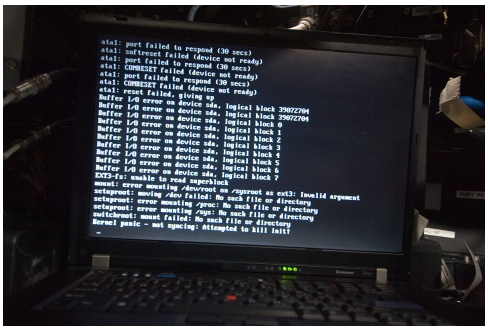

Image taken from commons.wikimedia.org.

In an earlier contest, contestants had to solve a simple problem. They were given a string and had to print each unique substring of it, along with the number of occurrences it had in the original string. For example `AB` would print `A 1 B 1 AB 1` and `AAA` would print `A 3 AA 2 AAA 1`.

When copying over this problem for reuse in this contest, several mistakes were made. The input constraints were changed significantly, making the problem absolutely impossible! Luckily this was partially cancelled out by the output validator being mangled as well. Now instead of checking for absolute correctness it only requires the number of occurrences of each character in the output to be correct. With a quick hotfix of applying run length encoding to the output, the problem was finally solvable again and the contest could continue on as planned. Right?

Input

The input contains a single string of length at least 1 and at most 10^6. It contains only ASCII upper and lower case characters. This string is then followed by a single newline character.

Output

For each non-whitespace character that appears a non-zero number of times in the output of the problem described above, print it along with its number of occurrences on a single line, separated by a space. Print the lines ordered by ascending values of the ASCII characters.

Sample Input 1	Sample Output 1

`ABC`	`1 6`
	`A 3`
	`B 4`
	`C 3`

Sample Input 2	Sample Output 2

`aaaab`	`1 6`
	`2 1`
	`3 1`
	`4 1`
	`a 20`
	`b 5`