Cantor

[The original version of this problem (in Spanish) can be found at http://www.dc.uba.ar/events/icpc/download/problems/taip2012-problems.pdf]

The mathematician Georg Cantor was a lover of both sets and infinity, but he didn't get along too well with his colleagues. One morning he woke up with the idea of defining a set so strange that, when made public, would make the rest of the mathematicians lose their sleep for several days. And he was successful.

The set he defined is called the Cantor set, and it is formed by all the real numbers in the interval **[0, 1]** whose decimal expression in base **3** uses exclusively the digits **0** and **2**. This set has amazing properties, which we will not mention here so that you can sleep tonight. Moreover, and luckily for everyone involved, in this problem we will not be working with the Cantor set, but with a generalization of this set to the integer numbers.

We will say that an integer number is of Cantor type, or a *cantiger* for short, if its expression in a given base **B** uses solely the digits in a given set **C** contained in **{0, 1, ..., B-1}**. Thus, the fact that a given number is a cantiger depends on how we choose **B** and **C**.

Your task is to count cantiger numbers, in order to prevent the mathematicians of the entire world from loosing their sleep. More precisely, given two integers **D** and **H**, along with **B** and **C**, you have to count the number of cantigers with respect to **B** and **C** from **D** to **H** inclusive.

Input

Each test case is described using a single line. This line contains three integers, **D**, **H** and **B**, and a string **L**. The values of **D** and **H** indicate the endpoints of the closed interval **[D, H]** we are interested in ($1 \le D \le H \le 10^{16}$). The value of **B** is the base mentioned in the problem statement ($2 \le B \le 10$). The string $L = L_0 L_1 ... L_{B-1}$ has exactly **B** characters, and describes the set **C** also mentioned in the problem statement. The character L_i is the uppercase letter **'S'** if **i** is in **C**, and the uppercase letter **'N'** otherwise (**i = 0, 1, ..., B-1**). The set **C** is non-empty, so that there is at least one **'S'** character in **L**. The end of the input is signalled by a line containing three times the number **-1** and a single **'*'** character.

Output

For each test case, you should print a single line containing an integer number, representing the number of cantigers (with respect to **B** and **C**) that are greater or equal to **D** and lower or equal to

H.

Example