D. Prefixes and Suffixes

time limit per test: 1 second
memory limit per test: 256 megabytes
input: standard input
output: standard output

You have a string $s = s_1 s_2 \ldots s_{|s|}$, where $|s|$ is the length of string s, and s_i its i-th character.

Let's introduce several definitions:

- A substring $s[i..j]$ $(1 \le i \le j \le |s|)$ of string s is string $s_i s_{i+1} \ldots s_j$.
- The prefix of string s of length l $(1 \le l \le |s|)$ is string $s[1..l]$.
- The suffix of string s of length l $(1 \le l \le |s|)$ is string $s[|s| - l + 1..|s|]$.

Your task is, for any prefix of string s which matches a suffix of string s, print the number of times it occurs in string s as a substring.

Input

The single line contains a sequence of characters $s_1 s_2 \ldots s_{|s|}$ $(1 \le |s| \le 10^5)$ — string s. The string only consists of uppercase English letters.

Output

In the first line, print integer k $(0 \le k \le |s|)$ — the number of prefixes that match a suffix of string s. Next print k lines, in each line print two integers l_i c_i. Numbers l_i c_i mean that the prefix of the length l_i matches the suffix of length l_i and occurs in string s as a substring c_i times. Print pairs l_i c_i **in the order of increasing** l_i.

Examples

input
ABACABA

output
3 1 4 3 2 7 1

input
AAA

output
3 1 3 2 2 3 1