

Suffix Array

Given a string of length at most 100,000 consist of alphabets and numbers. Output the suffix array of the string.

A [suffix array](#) is an array of integers giving the starting positions (0-based) of suffixes of a string in lexicographical order. Consider a string "abracadabra0AbRa4Cad14abra". The size of the suffix array is equal to the length of the string. Below is the list of 26 suffixes of the string along with its starting position sorted in lexicographical order:

POS SUFFIX

```
11 0AbRa4Cad14abra
20 14abra
16 4Cad14abra
21 4abra
12 AbRa4Cad14abra
17 Cad14abra
14 Ra4Cad14abra
25 a
10 a0AbRa4Cad14abra
15 a4Cad14abra
22 abra
7 abra0AbRa4Cad14abra
0 abracadabra0AbRa4Cad14abra
3 acadabra0AbRa4Cad14abra
18 ad14abra
5 adabra0AbRa4Cad14abra
13 bRa4Cad14abra
23 bra
8 bra0AbRa4Cad14abra
1 bracadabra0AbRa4Cad14abra
4 cadabra0AbRa4Cad14abra
19 d14abra
6 dabra0AbRa4Cad14abra
24 ra
9 ra0AbRa4Cad14abra
2 racadabra0AbRa4Cad14abra
```

Note: this is a partial score problem.

$O(n^2 \log(n))$ is expected to score about 20-30. (Naive sorting all suffixes)

$O(n \log^2(n))$ is expected to score about 40. (OK for most programming contest problems)

$O(n \log n)$ is expected to score about 60-70. (Use counting sort for small alphabet size)

$O(n)$ without tweaks is expected to score about 80-90.

$O(n)$ with tweaks is expected to score 100. (This is meant for fun only :)

Input

A single line containing the string.

Output

The suffix array of the string.

Example

Input:

abracadabra0AbRa4Cad14abra

Output:

11
20
16
21
12
17
14
25
10
15
22
7
0
3
18
5
13
23
8
1
4
19
6
24
9
2