Problem Statement (DecodeDigits)

A simple way to encode a word into a string of digits is to replace each letter by its order in the alphabet. That is, "a" will change to "1", "b" to "2", ..., and "z" to "26". For example, encode("cow")="31523" and encode("cat")="3120".

Sadly, this encoding cannot always be uniquely decoded, because two different words can yield the same string of digits when encoded. For example, encode("beard")=encode("yard")="251184".

String A is a subsequence of string B if it is possible to erase some letters of B (possibly none, possibly all of them) to obtain A. For example, "cage" is a subsequence of "cabbages".

You are given a String **D** containing a string of digits. If there is no string Y such that encode(Y)=**D**, return the String "`NONE`". Otherwise, find and return the longest string X with the following property: whenever encode(Y)=**D**, X is a subsequence of Y. If there are multiple such strings, return the lexicographically smallest one.

Definition

Class:	DecodeDigits
Method:	solve
Parameters:	String
Returns:	String
Method signature:	String solve(String D)

(be sure your method is public)

Constraints

-**D** will contain between 1 and 50 characters, inclusive.

-Each character in **D** will be a digit ('0'-'9').

Examples

0)
```
"38956"
```

```
Returns: "chief"
```
There is only one way to decode this string of digits.

1)
```
"13919156"
```

```
Returns: "if"
```
This string of digits can be decoded in 8 different ways. Each of them contains an "`i`" followed by an "`f`".

2)
```
"1122"
```

```
Returns: ""
```
We have encode("`kbb`")=encode("`aav`")="`1122`". The strings "`kbb`" and "`aav`" have no common subsequence other than the empty one.

3)

```
"3120"

Returns: "cat"
```

The only valid decoding of this string is `"cat"`.

4)

```
"0"

Returns: "NONE"
```