

## A. Interview with Oleg

time limit per test: 1 second

memory limit per test: 256 megabytes

input: standard input

output: standard output

Polycarp has interviewed Oleg and has written the interview down without punctuation marks and spaces to save time. Thus, the interview is now a string  $s$  consisting of  $n$  lowercase English letters.

There is a filler word `ogo` in Oleg's speech. All words that can be obtained from `ogo` by adding `go` several times to the end of it are also considered to be fillers. For example, the words `ogo`, `ogogo`, `ogogogo` are fillers, but the words `go`, `og`, `ogog`, `ogogog` and `oggo` are not fillers.

The fillers have maximal size, for example, for `ogogooo` speech we can't consider `ogo` a filler and `goo` as a normal phrase. We should consider `ogogo` as a filler here.

To print the interview, Polycarp has to replace each of the fillers with three asterisks. Note that a filler word is replaced with exactly three asterisks regardless of its length.

Polycarp has dealt with this problem in no time. Can you do the same? The clock is ticking!

### Input

The first line contains a positive integer  $n$  ( $1 \leq n \leq 100$ ) — the length of the interview.

The second line contains the string  $s$  of length  $n$ , consisting of lowercase English letters.

### Output

Print the interview text after the replacement of each of the fillers with `"***"`. It is allowed for the substring `"***"` to have several consecutive occurrences.

### Examples

input
7 aogogob
output
a***b

input
13 ogogmgogogogo
output
***gmg***

input
9 ogoogoogo
output
*****

### Note

The first sample contains one filler word `ogogo`, so the interview for printing is `"a***b"`.

The second sample contains two fillers `ogo` and `ogogogo`. Thus, the interview is transformed to `"***gmg***"`.