String insertion P21174_en

Implement an efficient data structure to keep a dynamic array `A[0..]` of strings, with two operations:

- '`I`' *s i*: Increase the size of *A* by one (like `A.push_back("");`). Move every string at a position *j* such that $j \geq i$ one position to its right. Store the string *s* at the *i*-th position, which now is empty.
- '`C`' *j*: Print the *j*-th character (0 based) of the whole array, considering the concatenation of all its strings from left to right.

Input

Input consists of just one case. Assume that each *s* has between 1 and 10 lowercase letters, each *i* is between 0 and the current number of strings, and each *j* is between 0 and the current number of characters minus one. The total number of operations is at most $3 \cdot 10^5$. An '`E`' marks the end of the input.

Output

Print a line with the letter at the *j*-th position for each '`C`' operation.

Sample input 1

```
I hello 0
C 0
C 4
I bye 0
C 0
C 7
I hi 1
C 4
C 1
C 9
E
```

Sample output 1

```
hoboiyo
```

Problem information

Author: Salvador Roura

Generation: 2026-01-25T10:10:59.871Z