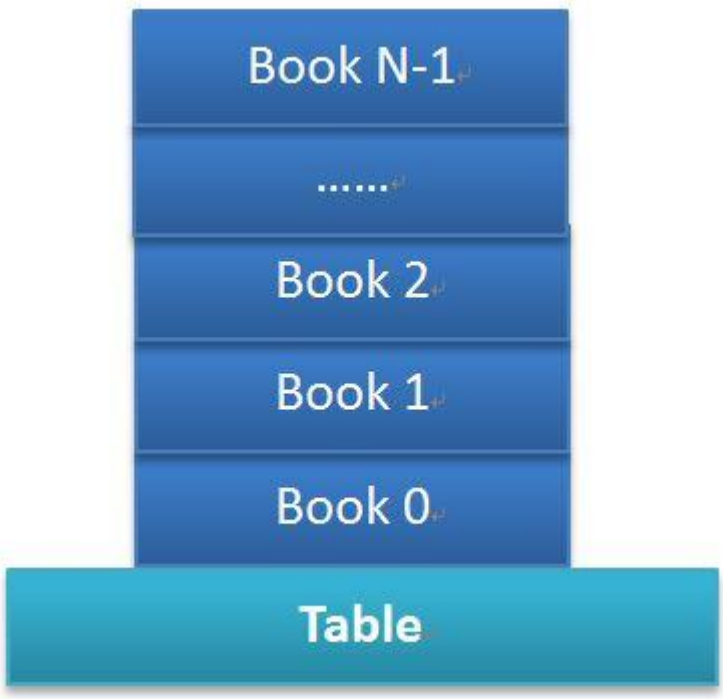
Description

The problem is to parse a series of commands to move the books that lie on the table. Initially there are n books lying on the table with book b_i adjacent to book b_{i+1} for all $0 \leq i < n - 1$,as shown in the diagram below:



The valid commands and limited for moving books are:

Illegal commands:

- A = B
- A or B is **not** in the range (e.g. You cannot move or remove any book that does not exist)

All illegal commands should be ignored and should have no affect on the configuration of books.

Valid commands:

move A on B

Puts book A on book B.

As below:

0 1 2 3 4

>> move 1 on 3

0 2 3 1 4

move A under B

Puts book A under of book B.

As below:

0 1 2 3 4

>> move 1 under 3

0 2 1 3 4

remove A

Remove book A from the list.

As below:

0 1 2 3 4

>> remove 1

0 2 3 4

exit

Finish moving the books, print the final status.

Input

The input begins with an integer n on a line by itself representing the number of books in the book world. You may assume that 0 < n <= 10000.

The number of books is followed by a sequence of book commands, one command per line. Your program should process all commands until the **exit** command is encountered.

You may assume that all commands will be of the form specified above. There will be no syntactically incorrect commands.

Output

Your output should contains one line of sequence which represents the order of books from the bottom to the top.
Each number is followed by a single space. And you are asked to add a new line character at the end.

Sample Input

```
5move 0 on 3move 2 under 0move 1 on 4remove 0exit
EOF
```

Sample Output

```
3 2 4 1
EOF
```