DSA Lab 6 Set 2 | Associativity

Input file:	`standard input`
Output file:	`standard output`
Time limit:	3 seconds
Memory limit:	1024 megabytes

For a given mathematical expression in infix notation with operands as positive integers and operators belonging to the set $\{+, -, *, /, (,)\}$ and the precedence rules are as stated below:

1. '(' when it is in the expression
2. $*$, $/$
3. $+$, $-$
4. '(' when it is inside the stack

Your task is to obtain the corresponding postfix notation such that for operators of equal precedence, the conversion algorithm follows a **right associativity**.
For instance, for an expression of the form '`a op1 b op2 c`', where a, b and c are operands and $op1$ and $op2$ are operators of equal precedence then the expression would be evaluated as '`a op1 (b op2 c)`' giving postfix notation of '`a b c op2 op1`'

Input

The first line contains an integer N, indicating the number of tokens. The second line contains N space-separated tokens representing the infix notation of the expression.

Constraints:
Basic:
$1 \leq N \leq 50$
tokens $\in \{1, 2, 3, 4, 5, 6, 7, 8, 9, 0, (,), +, -, *, /\}$
Advanced:
$1 \leq N \leq 100000$
tokens $\in \{1, 2, 3, 4, 5, 6, 7, 8, 9, 0, (,), +, -, *, /\}$

Output

N single space-separated tokens representing the postfix notation of the expression.

Note

Sample Test Case:
Input:
```
13
7 * 4 * 3 / ( 1 + 5 ) - 4
```
Output:
```
7 4 3 1 5 + / * * 4 -
```
Explanation:
Evaluation is done in this order:
$(1 + 5) \rightarrow 3/(1 + 5) \rightarrow 4 * (3/(1 + 5)) \rightarrow 7 * (4 * (3/(1 + 5))) \rightarrow (7 * (4 * (3/(1 + 5)))) - 4$