

3089 Ambiguous Expressions

A mathematics professor is trying to teach her students arithmetic involving large numbers. She writes down a big arithmetic expression on the board, using positive decimal numbers with arbitrary number of digits, and the binary addition, subtraction and multiplication operators. The operators are “infix” (i.e. one argument on the left, one on the right). However, she does not put any parenthesis in the expression, thus making it ambiguous. She announces that the first student who parenthesizes it in such a way that the value of the resulting expression is maximum, will get a prize.

Instead of wasting time adding and multiplying numbers, why not write a program to do it. That is exactly what you have to do.

Input

The input will contain several test cases. The first line of input will be a positive integer N , the number of test cases. The next N lines will contain a character string of at most 200 characters, describing the expression for each case, one per line. Each character is either a decimal digit or one of the binary operators ‘+’, ‘-’, ‘*’. The number of operators in each expression is at most 100. All the numbers are positive and there are no leading zeroes.

Output

The output should be a character string giving the maximum value of the expression, with the most significant digit first, and no leading zeroes, preceded by ‘-’ if the value is negative. If the value is 0, a single character ‘0’ should be printed.

The outputs for successive test cases should be printed one per line.

Sample Input

```
2
5-3*4-2*3
3-3*5
```

Sample Output

```
18
0
```