# 不喜欢括号

## 题目描述

NowCoder从小就喜欢数学，喜欢在笔记里记录很多表达式。它觉得现在的表达式写法很麻烦，为了提高运算符优先级，不得不添加很多括号，不小心漏了一个右括号的话差之毫厘谬之千里。

因此他改用前缀表达式，例如`(2 + 3) \* 4`写成`\* + 2 3 4`，这样就能避免使用括号了。这样的表达式书写简单，但计算却不够直观。请你写一个程序帮他计算这些前缀表达式吧。

### 输入描述:

输入包含多组数据，每组数据包含两行。第一行为正整数n（3≤n≤50），紧接着第二行包含n个由数值和运算符组成的列表。“+-\*/”分别为加减乘除四则运算，其中除法为整除，即“5/3=1”。

### 输出描述:

对应每一组数据，输出它们的运算结果。

### 输入例子:

3

+ 2 3

5

\* + 2 2 3

5

\* 2 + 2 3

### 输出例子:

5

12

10

## 解题思路

因为输入的是前缀式（也称逆波兰式），可以用一个栈来存储输入的操作数。**从输入序列的尾部向前进行处理**，当遇到操作数，就将操作数存入栈中；当遇到操作符时，从栈中弹出两个操作数进行计算，所得结果再次存入栈中。循环上面的操作直到所有的内容都处理完，最后栈中只有一个元素，这个元素就是所求的结果。