Brackets

This question is graded for 1%.

Problem Statement

Tom is a math enthusiast, and he recently learned of a succinct representation of multiplication and addition operations: the *bracket sequence*.

Normally, a series of arithmetic computations with $+$ and $\times$ can be represented as shown: $(10 \times (1 + 2 + 3 + 4) \times 5) + 3 + 2$. The brackets are only utilised to group multiplications and additions together. Hence, all the operators can be eliminated, with noting that addition is used for numbers outside any parentheses. The sequence can be then shorted to its bracket sequence form: $(10\ (1\ 2\ 3\ 4)\ 5)\ 3\ 2$.

To perform computations on a bracket sequence, it is important to highlight that each pair of parentheses can only group one operation, and for nested brackets, the addition and multiplication operations alternate. $()$ is not a valid bracket sequence and it is not a subsequence of any valid bracket sequence.

Can you help Tom write a program to evaluate any valid bracket sequences?

Input

The first line of input consists of a single integer N ($1 <= N <= 3 \times 10^5$), the number of tokens. The second line of input consists of N space separated tokens where each token is either (,) or an integer c ($0 <= c < 10^9 + 7$). The tokens are guaranteed to form a valid bracket sequence.

Output

Compute the value of the bracket sequence, modulo $10^9 + 7$. Note that it is advisable to perform the necessary modulo operations at every step, and avoid usage of the BigInteger class.

Sample Input 1

```
2
10 22
```

Sample Output 1

```
32
```

Sample Input 2

```
8
( 3 ( 2 5 ) ) 3
```

Sample Output 2

```
24
```

Sample Input 3

```
4
( 4 5 )
```

Sample Output 3

```
20
```

Sample Input 4

```
6
( 4 ) ( 5 )
```

Sample Output 4

```
9
```

Sample Input 5

```
9
( 5 ) ( 7 ) ( 9 )
```

Sample Output 5

```
21
```

Sample Input 6

```
6
( ( 3 4 ) )
```

Sample Output 6

```
7
```

Explanation

For Sample Input 2, (3 (2 5)) 3 $= \big(3 \times (2 + 5)\big) + 3 = 24$

Grading Bands:

- If your approach solves the question in $O\ (N\ log\ N)$ time or better, you will attain the full 100%.
- If your approach solves the question in $O(N^2)$ time or better, you will attain ~~50%~~ 0%.
- If your approach solves the question in $O(N^3)$ time or better, you will attain ~~10%~~ 0%.