Asterisk Expressions

by pkacprzak

Problem Submissions Leaderboard Discussions

In Mathematics, *exponentiation* has higher operator precedence than multiplication. This means that any exponentiation in the expression must be evaluated first before before any multiplication. For the purposes of this challenge, exponentiation is left-associative, which is different than you would normally evaluate it. This means that 2^{3^2} is evaluated as $(2^3)^2 = 64$ and *not* as $2^{(3^2)} = 512$.

In Python, the exponentiation operator is denoted by a double asterisk (`**`). For example, a^k is expressed as `a**k`. To express multiplication, we use a single asterisk operator, `*`. For example, we express $a \times b$ as `a*b`.

An expression, e, consisting of decimal digits and asterisks is *valid* if and only if it forms a valid mathematical expression when each double-asterisk (i.e., `**`) is replaced with a math exponentiation sign. For example, `2**4*5**3` is a valid expression translating to $2^4 \times 5^3$, while `*2**3`, `2***3`, `4*5**`, and `4**2*` are invalid as they do not translate to valid mathematical expressions.

Given t expressions consisting of decimal digits and asterisks, parse each expression and determine its validity. If an expression is valid, print its evaluated value modulo $10^9 + 7$ on a new line; if it's invalid, print `Syntax Error` instead.

Input Format

The first line contains a single integer, t, denoting the number of expressions.
Each line i of the t subsequent lines contains a single string, e_i, denoting an expression.

Constraints

- $1 \leq t \leq 10$
- $1 \leq |e| \leq 10^5$
- There are at most 17 consecutive digits in an expression.
- Consecutive sequences of digits in an expression will never start with 0.
- Each expression e consists of decimal digits (i.e., 0 through 9) and asterisks (`*`) only.

Output Format

For each expression, e_i, print its answer on a new line. If e_i is valid, the answer is the evaluated expression $\bmod (10^9 + 7)$; otherwise, it's `Syntax Error`.

Sample Input

Sample Input 0

```
1
3*2**3**2*5
```

Sample Output 0

```
960
```

Explanation 0
We have $n = 1$ expression to evaluate. Because exponentiation has higher operator precedence than multiplication, and because exponentiation is left-associative here, the expression evaluates to $3 \times (2^3)^2 \times 5 = 3 \times 64 \times 5 = 960 \bmod (10^9 + 7) = 960$. Thus, we print 960 on a new line.

Sample Input 1

```
1
3***4
```

Sample Output 1

```
Syntax Error
```

Explanation 1

We have $n = 1$ expression to evaluate. Because this expression contains three consecutive asterisks, it cannot be evaluated as a mathematical expression. Thus, we print *Syntax Error* on a new line.

Contest ends in a day

Submissions: 933

Max Score: 50

Difficulty: Medium

Rate This Challenge:

More

Current Buffer (saved locally, editable)

Python 3

1

Upload Code as File

Test against custom input

Run Code

Submit Code