

## Задача E. Большое дерево

Имя входного файла: E.in  
Имя выходного файла: E.out  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 64 мегабайта

Вам дано дерево из  $N$  вершин. Дерево — это неориентированный граф, между любыми двумя вершинами которого, существует ровно один путь. Необходимо достроить его, добавив к нему  $M$  новых вершин, и  $M$  новых ребер так, чтобы произведение степеней всех вершин было максимальным, и чтобы получившийся граф оставался деревом.

### Формат входного файла

На первой строке входных данных два целых числа  $N, M$ . Далее в  $N - 1$  строках заданы ребра дерева, по два различных целых числа  $1 \leq a, b \leq N$  - номера вершин, соединенных ребром.

### Формат выходного файла

Выведите максимально возможное произведение степеней всех вершин в полученном дереве по модулю 1000000007.

### Примеры

E.in	E.out
2 1 1 2	2
2 0 1 2	1

### Примечание

Степень вершины - количество ребер, соединяющихся в ней.

## Задача F. Арифметика на отрезках

Имя входного файла: F.in  
Имя выходного файла: F.out  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 64 мегабайта

Дана последовательность из  $N$  целых неотрицательных чисел. Ваша задача заключается в том, чтобы уметь выполнять три вида запросов:

1. + l r d – прибавить ко всем числам на отрезке от  $l$  до  $r$  число  $d$  ( $1 \leq l \leq r \leq N, 0 \leq d \leq 10^9$ );
2. \* l r d – умножить все числа на отрезке от  $l$  до  $r$  на число  $d$  ( $1 \leq l \leq r \leq N, 0 \leq d \leq 10^9$ );
3. ? p – вывести значение  $p$ -го элемента в этой последовательности по модулю  $10^9 + 7$  ( $1 \leq p \leq N$ ).

### Формат входного файла

В первой строке входного файла задано целое число  $N$ . В следующей строке задано  $N$  целых неотрицательных чисел, элементы последовательности. Каждое из этих чисел не превосходит  $10^9$ . В третьей строке задано число  $M$  — количество запросов. В последующих  $M$  строках заданы запросы, соответствующие описанию из условия.

### Формат выходного файла

Для каждого запроса третьего вида выведите ответ на одной строке, в том порядке, в котором заданы запросы.

### Примеры

F.in	F.out
4	9
1 3 0 7	51
5	
+ 3 3 9	
? 3	
* 2 4 6	
+ 1 4 9	
? 4	

## Задача G. Счастливые перестановки

Имя входного файла: G.in  
Имя выходного файла: G.out  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 64 мегабайта

Назовем число, состоящее из  $2N$  или  $2N + 1$  цифр счастливым, если сумма первых  $N$  цифр равна сумме последних  $N$  цифр. Вам дан набор из  $M$  карточек, на каждой из которых записано  $K$  цифр. Сколькими способами можно расставить все эти карточки в ряд так, чтобы полученное в результате число было счастливым?

### Формат входного файла

На первой строке записаны числа  $M$  и  $K$ . На следующих  $M$  строках записаны последовательности, состоящие из  $K$  цифр каждая.

### Формат выходного файла

Выведите одно число — ответ к задаче.

### Примеры

G.in	G.out
4 1 1 2 3 4	8
2 2 11 11	2

## Задача Н. Сравнение подстрок

Имя входного файла: Н.in  
Имя выходного файла: Н.out  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 64 мегабайта

Дана строка  $S$ . Каждой паре чисел  $(l, r)$ , удовлетворяющих условию  $1 \leq l \leq r \leq |S|$ , соответствует подстрока строки  $S$ , начинающаяся в позиции  $l$  и заканчивающаяся в позиции  $r$  (включительно). У вас есть два типа запросов:

- удалить символ в позиции  $i$ , после чего оставшиеся части строки, если такие есть, склеиваются (гарантируется, что такая позиция существует)
- определить, равны ли подстроки  $(l_1, l_1 + len - 1)$  и  $(l_2, l_2 + len - 1)$ .

### Формат входного файла

В первой строке дана строка  $S$ , которая состоит только из 0 и 1. Во второй строке записано целое число  $N$  — количество запросов. Далее идут  $N$  строк, в каждой из которых описание запроса:

- $i$  — запрос типа 1
- ?  $l_1$   $l_2$   $len$  — запрос типа 2

### Формат выходного файла

Для каждого запроса типа 2, вывести “а”, если подстроки совпадают, или “jok” в противном случае.

### Примеры

Н.in	Н.out
001001	ia
5	jok
? 1 4 2	ia
? 1 2 3	jok
- 4	
? 2 4 2	
? 1 3 1	
010110	ia
4	
- 1	
- 1	
- 1	
? 1 2 1	