

## Задача А. Посланник

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	5 секунд
Ограничение по памяти:	256 мегабайт

За долгие годы правления король Байтс так устал, что решил отречься от престола. Но достаточно скоро он понял, что упускает последние новости и интриги. Поэтому, чтобы оставаться в курсе, он решил стать королевским посланником.

В первый же день его новой работы ему было поручено доставить срочное сообщение из одного города в другой. Вместо того, чтобы сделать свою работу как можно быстрее, он решил устроить тур по стране. Естественно, чтобы новый король этого не узнал, ему нужно принять некоторые меры предосторожности.

Все дороги между парами городов в стране являются односторонними. Байтс указал точное количество дорог, которые он собирается проехать, независимо от того, сколько действительно требуется. Чтобы не вызывать никаких подозрений, он хочет посетить исходный город и город назначения ровно один раз. Любой другой город и дороги он может посещать несколько раз.

Помогите бывшему королю, написав программу, которая определит количество маршрутов, удовлетворяющих его требованиям.

Другими словами, вам нужно определить количество различных маршрутов заданной длины между данной парой городов, каждый из которых можно посетить ровно один раз. И поскольку это число может быть довольно большим, достаточно вычислить его остаток от деления на число, которое выбрал Байтс.

### Формат входных данных

В первой строке задано три числа  $n$ ,  $m$  и  $z$  ( $2 \leq n \leq 100$ ,  $0 \leq m \leq n(n-1)$ ,  $2 \leq z \leq 10^9$ ) — количество городов, количество дорог и число, которое выбрал Байтс. Города пронумерованы от 1 до  $n$ .

Каждая из следующих  $m$  строк содержит целые числа  $a$  и  $b$  ( $1 \leq a, b \leq n$ ,  $a \neq b$ ), что означает существование дороги из города  $a$  в город  $b$ . Никакая дорога не встречается более одного раза.

В следующей строке дано число  $q$  ( $1 \leq q \leq 500\,000$ ) — количество запросов Байтса. Каждая из следующих  $q$  строк содержит описание запроса. Запрос содержит три целых числа  $u_i$ ,  $v_i$  и  $d_i$  ( $1 \leq u_i, v_i \leq n$ ,  $u_i \neq v_i$ ,  $1 \leq d_i \leq 50$ ) — номера начального и конечного города маршрута и количество дорог в нем.

### Формат выходных данных

Выведите  $q$  строк.  $i$ -я строка должна содержать остаток от деления на  $z$  числа маршрутов для  $i$ -го запроса.

### Система оценки

Подзадача 1 (12 баллов):  $n \leq 20$ ;  $q \leq 100$

Подзадача 2 (20 баллов):  $n \leq 100$ ;  $m \leq 500$ ;  $q \leq 100$

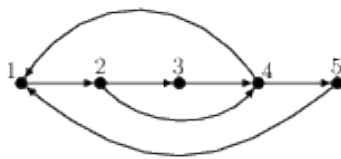
Подзадача 3 (38 баллов):  $n \leq 100$ ;  $q \leq 10\,000$

Подзадача 4 (30 баллов): Нет дополнительных условий.

## Пример

стандартный ввод	стандартный вывод
5 7 10	2
1 2	1
2 3	
3 4	
4 5	
5 1	
2 4	
4 1	
2	
2 1 3	
5 3 6	

## Замечание



**Пояснения к примеру:** Два маршрута удовлетворяют первому запросу:  $2 \rightarrow 3 \rightarrow 4 \rightarrow 1$  и  $2 \rightarrow 4 \rightarrow 5 \rightarrow 1$ ; и только один маршрут удовлетворяет второму запросу:  $5 \rightarrow 1 \rightarrow 2 \rightarrow 4 \rightarrow 1 \rightarrow 2 \rightarrow 3$ .

## Задача В. Трудолюбивый Джонни

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1.5 секунд
Ограничение по памяти:	256 мегабайт

Сегодня День Рождения маленького Джонни. Бедный ребенок не получил никаких игрушек, ни игр, ни компьютера на свой день рождения. Вместо этого ему подарили длинные массивы чисел, деревья, карты странных мест с дорогами, которые проходят через многочисленные туннели и эстакады, длинные ленты заполненные 1048576 символами префиксов Фибоначчи и Туэ-Морса, и т.д. Из всех этих образовательных подарков ему больше всего понравилась перестановка<sup>1</sup> из первых  $n$  положительных целых чисел. Вскоре Джонни начал задаваться вопросом, какая была лексикографически предыдущая перестановка<sup>2</sup> для той, что ему дали. Выяснив это довольно быстро, Джонни сразу спросил себя, как он может записать эту предшествующую перестановку в свой массив. Единственной операцией, поддерживаемой массивом, является выбор двух ячеек и заменой чисел в этих ячейках. К счастью, Джонни был достаточно умен, чтобы преобразовать начальную перестановку к предыдущей за минимальное количество обменов. Он нашел эту задачу настолько увлекательной, что продолжал преобразовывать каждую полученную перестановку в предыдущую.

В своем перестановочном безумии, Джонни совершенно позабыл о гостях, чем вначале забавлял их, но вскоре и расстроил. Один из гостей понял, что Джонни остановится, когда перейдет к перестановке  $1, 2, \dots, n$ , которая является лексикографически минимальной. Сколько же времени это займет?

Помогите им ответить на этот вопрос, зная, что каждый обмен между ячейками массива занимает ровно одну секунду. Поскольку это может занять достаточно много времени (Трудяга — второе имя Джонни), им будет достаточно знать остаток от деления на  $10^9 + 7$ . В конце концов, они могут проверять Джонни каждые  $10^9 + 7$  секунд, закончил он или нет.

### Формат входных данных

В первой строке находится одно целое число  $n$  — длина перестановки, которую получил Джонни на свой день рождения ( $1 \leq n \leq 1\,000\,000$ ).

Во второй строке задана перестановка из  $n$  элементов — попарно различные целые числа  $p_1, p_2, \dots, p_n$  разделенные пробелами ( $1 \leq p_i \leq n$ ).

### Формат выходных данных

Выведите количество обменов по модулю  $10^9 + 7$ , которые должен сделать Джонни, чтобы перестановка стала минимальной.

### Система оценки

Подзадача 1 (15 баллов):  $n \leq 10$

Подзадача 2 (37 баллов):  $n \leq 5\,000$

Подзадача 3 (15 баллов):  $n \leq 1\,000\,000$ ; перестановка имеет следующий вид:  $n, n-1, \dots, 1$

Подзадача 4 (33 баллов): Нет дополнительных ограничений.

### Пример

стандартный ввод	стандартный вывод
3 3 1 2	6

### Замечание

Из заданной в тестовом примере перестановки Джонни получит следующие:  $(2, 3, 1)$ ,  $(2, 1, 3)$ ,

<sup>1</sup>Перестановка чисел от 1 до  $n$  — это последовательность попарно различных чисел  $p_1, \dots, p_n$ , удовлетворяющих  $1 \leq p_i \leq n$  (каждое число от 1 до  $n$  появляется в перестановке только один раз).

<sup>2</sup>Перестановка  $P = (p_1, \dots, p_n)$  лексикографически меньше перестановки  $Q = (q_1, \dots, q_n)$  (которые обозначим  $P < Q$ ) Если  $p_j < q_j$ , где  $j$  это наименьший индекс, в котором  $p_j \neq q_j$ . Перестановка  $P$  лексикографически предыдущая для  $Q$ , если  $P < Q$  и не существует перестановки  $R$ , что  $P < R < Q$ .

$(1, 3, 2), (1, 2, 3)$ . Чтобы получить их, ему придется сделать в общей сложности  $2 + 1 + 2 + 1 = 6$  замен.

## Задача С. Распределение билетов

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 8 секунд  
Ограничение по памяти: 256 мегабайт

В Байтсити построили кольцевое метро, в котором  $n$  станций пронумерованных от 1 до  $n$ . Они расположены по порядку по часовой стрелке.

Введены  $n$  типов билетов также пронумерованных от 1 до  $n$ . С помощью одного билета типа  $i$  ( $1 \leq i \leq n-1$ ), можно проехать от станции с номером  $i$  до станции с номером  $i+1$ , либо от станции с номером  $i+1$  до станции с номером  $i$ . С помощью одного билета типа  $n$ , можно проехать от станции с номером  $n$  до станции с номером 1, либо от станции с номером 1 до станции с номером  $n$ . Билеты в метро продаются только наборами из  $n$  билетов по одному каждого типа.

Вы работаете в туристическом агентстве и ваша задача организовать оптимальную покупку билетов.

На сегодня зарегистрировано  $m$  заказов. В  $i$ -м заказе группа из  $c_i$  туристов хочет проехать со станции с номером  $a_i$  до станции с номером  $b_i$ . Маршруты в одной группе туристов необязательно должны совпадать.

Требуется определить минимальное количество наборов билетов, которое нужно приобрести, чтобы обеспечить проезд для всех групп туристов.

### Формат входных данных

Первая строка входных данных содержит два целых числа  $n$  и  $m$  — количество станций в метро и количество заказов на сегодня, соответственно ( $3 \leq n \leq 200\,000$ ;  $1 \leq m \leq 100\,000$ ).

Каждая из следующих  $m$  строк содержит три целых числа  $a_i$ ,  $b_i$  и  $c_i$  — начальная станция, станция, на которую хотят попасть и количество туристов, зарегистрированных в заказе с номером  $i$ , соответственно ( $1 \leq a_i, b_i \leq n$ ;  $a_i \neq b_i$ ;  $1 \leq c_i \leq 10^9$ ;  $1 \leq i \leq m$ ).

### Формат выходных данных

Выведите единственное целое число — количество наборов билетов, которое нужно приобрести, чтобы обеспечить проезд для всех групп туристов.

### Система оценки

Подзадача 1 (10 баллов):  $n, m \leq 20$ ;  $c_i = 1$ .

Подзадача 2 (35 баллов):  $n, m \leq 300$ ;  $c_i = 1$ .

Подзадача 3 (20 баллов):  $n, m \leq 3\,000$ ;  $c_i = 1$ .

Подзадача 4 (20 баллов):  $c_i = 1$ .

Подзадача 5 (15 баллов): Нет дополнительных ограничений.

### Примеры

стандартный ввод	стандартный вывод
3 3 1 2 1 2 3 1 3 1 1	1
3 2 1 2 4 1 2 2	3
6 3 1 4 1 2 5 1 3 6 1	2

## Замечание

Для первого тестового примера достаточно купить один набор билетов, и тогда все смогут достичь цели, перемещаясь по часовой стрелке.

Для второго тестового примера достаточно купить три набора билетов, если туристы будут перемещаться следующим образом:

- Для первого заказа, трое туристов едут по часовой стрелке и один против часовой.
- Для второго заказа, оба туриста едут против часовой стрелки.

Невозможно выполнить заказы меньшим числом наборов билетов.

Для третьего тестового примера достаточно купить два набора билетов и распределить их следующим образом:

- Предоставить билеты с номерами 1, 2, 3 туристу, который хочет проехать со станции с номером 1 до станции с номером 4.
- Предоставить билеты с номерами 1, 6, 5 туристу, который хочет проехать со станции с номером 2 до станции с номером 5.
- Предоставить билеты с номерами 3, 4, 5 туристу, который хочет проехать со станции с номером 3 до станции с номером 6.

Невозможно выполнить заказы только с одним набором билетов.