

Задача А. Контейнеры: перезагрузка

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

В компании грузоперевозок Нурлаш и КО inc. для перевозок грузов используют контейнеры разных размеров. Когда нужда в них отсутствует, некоторые из них вкладывают в другие для экономии места. Когда контейнер становится нужным, специальный робот достает его и вынимает содержимое.

Вы главный разработчик в компании и конечно знаете что разборочный робот работает следующим образом : сначала он вытаскивает все содержимое контейнера который нужно разобрать и выписывает его номер. После он запускает процедуры разбора для контейнеров которые вынули, в том порядке в котором они находились внутри него.

В конце процедуры разбора у вас появляется список номеров и вам очень не нравится когда в нем встречается плохая пара чисел. Плохой парой последовательности $A[]$ будем называть такую пару i и j что $A[i] > A[j]$ причем $i < j$. Вы решили переписать робота упаковщика, но для начала вы хотите узнать, на самом ли деле все так плохо? Для этого вы решили подсчитать, для каждого контейнера, сколько плохих пар получится в выписанной последовательности.

Формат входных данных

Первая строка входных данных содержит единственное число — $N(1 \leq N \leq 2 * 10^5)$, количество контейнеров. Все контейнеры пронумерованы различными целыми числами от 1 до N .

Следующие N строк содержат информацию о контейнерах. В $i + 1$ строке содержится описание контейнера под номером i в виде $K_i(1 \leq K_i < N)$ и K_i чисел $V_i(1 < V_i \leq N, V_i \neq i)$. Где K_i это количество контейнеров находящихся внутри i -го, V_i это их номера, в том порядке в котором они лежат внутри.

Гарантируется что каждый контейнер на прямую лежит ровно в одном другом, кроме контейнера под номером 1. Также все контейнеры на прямую или через другие контейнеры лежат внутри контейнера под номером 1.

Формат выходных данных

В единственную строку входного файла выведите N чисел. Где i -ое число — количество плохих пар которые получатся если разборочный робот начнет разбирать контейнер под номером i .

Система оценки

Данная задача содержит пять подзадач:

1. $1 \leq N \leq 1000$, $K_1 = N - 1$. Подзадача оценивается в 10 баллов.
2. $1 \leq N \leq 2 * 10^5$, $K_1 = N - 1$. Подзадача оценивается в 20 баллов.
3. $1 \leq N \leq 500$. Подзадача оценивается в 10 баллов.
4. $1 \leq N \leq 5000$. Подзадача оценивается в 25 баллов.
5. $1 \leq N \leq 2 * 10^5$. Подзадача оценивается в 35 баллов.

Пример

стандартный ввод	стандартный вывод
3	1 0 0
2 3 2	
0	
0	

Задача В. Головоломка для Джека и Джилла

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Джек и Джилл братья любящие решать сложные головоломки. Однажды их мама подарила им M различных головоломок. Головоломки пронумерованы от 1 до M и сложность головоломок возрастает с увеличением его номера (т.е. головоломка под номером X сложнее головоломки под номером Y если $X > Y$).

В следующие N дней каждый из братьев будут решать ровно по одной головоломке в день. Они верят что каждый день они развиваются и каждый день будут решать более сложную головоломку, чем в предыдущий день. Более того, Джилл как старший брат считает себя умнее Джека и поэтому он будет решать более сложную головоломку чем его брат в этот же день. Формально все вышесказанное выглядит так: Пусть номера головоломок решенных Джеком a_1, a_2, \dots, a_N и номера решенных Джиллом b_1, b_2, \dots, b_N , тогда должны выполняться следующие условия:

- $a_i < a_{i+1}$ (для всех i от 1 до $N - 1$)
- $b_i < b_{i+1}$ (для всех i от 1 до $N - 1$)
- $a_i < b_i$ (для всех i от 1 до N)

Ваша задача состоит в том, чтобы посчитать количество способов братьям решать головоломки. Два способа считаются различными, если набор решенных головоломок одним из братьев не совпадают. Кроме того, во всех подзадачах кроме последнего есть условие *уникальности* - ни одна головоломка не может быть решена обоими братьями, т.е. последовательности a_1, a_2, \dots, a_N и b_1, b_2, \dots, b_N **не могут** содержать общих элементов. На последней подзадаче не выполняется условие *уникальности*, т.е. последовательности a_1, a_2, \dots, a_N и b_1, b_2, \dots, b_N **могут** содержать общих элементов.

Формат входных данных

В единственной строке входных данных содержится 3 целых положительных числа M, N, U — количество головоломок, количество дней и *уникальность* ($U = 1$ означает что условие *уникальности* выполняется, $U = 0$ означает что условие *уникальности* не выполняется)

Формат выходных данных

Выведите одно число — ответ на задачу по модулю $1000000007 (10^9 + 7)$.

Система оценки

Данная задача содержит 6 подзадач:

1. $1 < 2 * N \leq M \leq 10, U = 1$. Оценивается в 15 баллов.
2. $1 < 2 * N \leq M \leq 26, U = 1$. Оценивается в 10 баллов.
3. $1 < 2 * N \leq M \leq 500, U = 1$. Оценивается в 15 баллов.
4. $1 < 2 * N \leq M \leq 5000, U = 1$. Оценивается в 10 баллов.
5. $1 < 2 * N \leq M \leq 2 * 10^5, U = 1$. Оценивается в 35 баллов.
6. $1 < 2 * N \leq M \leq 500, U = 0$. Оценивается в 15 баллов.

Примеры

стандартный ввод	стандартный вывод
4 2 1	2
4 2 0	6

Задача С. Путешествие

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	32 мегабайта

Максат живет в стране где есть N городов. Города соединены $N - 1$ дорогами одинаковой длины. Из каждого города можно добраться до любого другого города. В каждом городе живет некоторое количество жителей. Максат каждую неделю выбирает два города и идет по кратчайшему пути от одного города до другого. В своем пути он выбирает два города, таких что общее количество жителей в этих двух городах равно C . Вы хотите посчитать сколькими способами он мог выбрать два города на этом пути таких, что общее количество жителей в этих двух городах равно C .

Формат входных данных

В первой строке дано количество городов N , во второй строке количество жителей в каждом городе $a_i (1 \leq a_i \leq 10^9)$.

В следующих $N - 1$ строках даны пары чисел, обозначающих города соединенные дорогой.

В следующей строке даны два числа — количество путешествий M и общее количество жителей в двух городах $C (1 \leq C \leq 10^9)$.

В следующих M строках даны пары городов в которых Максат начал и закончил свой путь.

Формат выходных данных

Для каждого путешествия выведите ответ.

Система оценки

Данная задача содержит 4 подзадачи:

1. $1 \leq N \leq 3, M = 1$. Оценивается в 10 баллов.
2. $1 \leq N \leq 300, 1 \leq M \leq 300$. Оценивается в 15 баллов.
3. $1 \leq N \leq 5000, 1 \leq M \leq 5000$. Оценивается в 25 баллов.
4. $1 \leq N \leq 50000, 1 \leq M \leq 50000$. Оценивается в 50 баллов.

Примеры

стандартный ввод	стандартный вывод
3 10 20 30 1 2 2 3 1 40 1 3	1
3 10 10 10 1 2 2 3 1 20 1 3	3

Задача D. Магазин «Всё за $O(1)$ »

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	512 мегабайт

В магазине «Всё за $O(1)$ » есть две кассы и много посетителей. Вам предстоит смоделировать очереди в эти кассы по записанной истории работы магазина.

Вам в хронологическом порядке даны события, закодированные следующими символами:

- a — в конец очереди в первую кассу встал очередной посетитель;
- b — в конец очереди во вторую кассу встал очередной посетитель;
- A — в первой кассе обслужили первого посетителя в очереди;
- B — во второй кассе обслужили первого посетителя в очереди;
- > — первая касса закрылась;
-] — вторая касса закрылась;
- < — первая касса открылась;
- [— вторая касса открылась.

Когда касса закрывается, все люди из очереди к этой кассе в обратном порядке, начиная с последнего, переходят в конец другой очереди. То есть первым переходит человек, стоявший последним, затем человек, стоявший предпоследним, и так далее. В итоге последним в получившейся очереди будет стоять тот, кто был первым в очереди к только что закрывшейся кассе.

Когда закрытая касса открывается, люди в очереди к другой кассе, начиная с последнего, переходят в нее, если их место в новой очереди окажется строго меньше текущего. Стоявший последним становится первым в новой очереди, стоявший предпоследним становится вторым и так далее.

Список событий корректен, то есть:

- Открываются только закрытые кассы;
- Закрываются только открытые кассы;
- Посетители не встают в очереди к закрытым кассам;
- Закрытые кассы не пытаются обслуживать посетителей;
- Кассы не обслуживают посетителей, если очереди к ним пустые;
- В каждый момент времени работает хотя бы одна касса.

Посетители нумеруются с единицы в порядке их появления в списке событий. В начальный момент обе кассы открыты и обе очереди пусты.

Формат входных данных

В первой строке входных данных содержится натуральное число n ($2 \leq n \leq 10\,000\,000$) — количество событий.

Во второй строке содержатся n символов, описывающих события согласно приведённым выше обозначениям.

Гарантируется, что во входных данных содержится хотя бы один запрос обслуживания посетителя

Формат выходных данных

В единственной строке выведите для каждой записи обслуживания последнюю цифру номера обслуженного посетителя. Ответы выводите в порядке выполнения запросов обслуживания, не используйте никаких разделителей.

Система оценки

Тесты к этой задаче состоят из трёх групп. Баллы за каждую группу ставятся только при прохождении всех тестов группы и всех тестов предыдущих групп.

- Группа, в которой $n \leq 1000$ оценивается в 26 баллов
- Группа, в которой $n \leq 200\,000$ оценивается в 37 баллов
- Группа, в которой $n \leq 10\,000\,000$ оценивается в 37 баллов

Примеры

стандартный ввод	стандартный вывод
15 aaabA>bBBb<BBAa	143256
12 aaaaa><AABBB	12543

Замечание

Пояснение к первому примеру:

№	Команда	Пояснение	1 очередь	2 очередь
1	a	В очередь 1 встал посетитель 1	1	–
2	a	В очередь 1 встал посетитель 2	1, 2	–
3	a	В очередь 1 встал посетитель 3	1, 2, 3	–
4	b	В очередь 2 встал посетитель 4	1, 2, 3	4
5	A	В очереди 1 обслужен посетитель 1	2, 3	4
6	>	Касса 1 закрылась	–	4, 3, 2
7	b	В очередь 2 встал посетитель 5	–	4, 3, 2, 5
8	B	В очереди 2 обслужен посетитель 4	–	3, 2, 5
9	B	В очереди 2 обслужен посетитель 3	–	2, 5
10	b	В очередь 2 встал посетитель 6	–	2, 5, 6
11	<	Касса 1 открылась	6	2, 5
12	B	В очереди 2 обслужен посетитель 2	6	5
13	B	В очереди 2 обслужен посетитель 5	6	–
14	A	В очереди 1 обслужен посетитель 6	–	–
15	a	В очередь 1 встал посетитель 7	7	–