

Задача А. Минимизируй!

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 3 секунды
Ограничение по памяти: 512 мегабайт

Дан массив целых чисел a длины n . Поступает q запросов двух типов:

- 1 $l r x$. Для каждого i на отрезке от l до r включительно нужно заменить a_i на $\min(a_i, x)$.
- 2 $l r$. Необходимо вывести сумму элементов массива a на отрезке от l до r включительно.

Формат входных данных

В первой строке входных данных дано целое число n ($1 \leq n \leq 300\,000$) — количество элементов массива a .

Во второй строке даны n целых чисел a_i ($1 \leq a_i \leq 10^9$) — элементы массива a .

В третьей строке дано целое число q ($1 \leq q \leq 300\,000$) — количество запросов.

В последующих q строках даны запросы по одному в строке.

Запрос первого типа задается так: 1 $l r x$

Где $1 \leq l \leq r \leq n$, $1 \leq x \leq 10^9$ — целые числа. Это означает, что все элементы массива a на отрезке от l до r нужно заменить на минимум из текущего значения и x .

Запрос второго типа задается так: 2 $l r$

Где $1 \leq l \leq r \leq n$ — целые числа. Это означает, что нужно вывести сумму элементов массива a на отрезке от l до r .

Формат выходных данных

Для каждого запроса второго типа выведите в отдельной строке сумму элементов на соответствующем отрезке.

Примеры

стандартный ввод	стандартный вывод
3	7
1 4 2	6
5	3
2 1 3	
1 1 3 3	
2 1 3	
1 1 3 1	
2 1 3	
7	118
1 7 2 4 8 4 100	117
7	9
1 3 6 3	17
2 2 7	
1 2 3 5	
2 1 7	
1 1 7 3	
2 1 4	
2 2 7	

Задача В. Минимизируй, прибавляй!

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	3 секунды
Ограничение по памяти:	512 мегабайт

Дан массив целых чисел a длины n . Поступает q запросов трех типов:

- $1\ l\ r\ x$. Для каждого i на отрезке от l до r включительно нужно заменить a_i на $\min(a_i, x)$.
- $2\ l\ r\ x$. Для каждого i на отрезке от l до r включительно нужно заменить a_i на $a_i + x$.
- $3\ l\ r$. Необходимо вывести сумму элементов массива a на отрезке от l до r включительно.

Формат входных данных

В первой строке входных данных дано целое число n ($1 \leq n \leq 300\,000$) — количество элементов массива a .

Во второй строке даны n целых чисел a_i ($-10^9 \leq a_i \leq 10^9$) — элементы массива a .

В третьей строке дано целое число q ($1 \leq q \leq 300\,000$) — количество запросов.

В последующих q строках даны запросы по одному в строке.

Запрос первого типа задается так: $1\ l\ r\ x$

Где $1 \leq l \leq r \leq n$, $-10^9 \leq x \leq 10^9$. Это означает, что все элементы массива a на отрезке от l до r нужно заменить на минимум из текущего значения и x .

Запрос второго типа задается так: $2\ l\ r\ x$

Где $1 \leq l \leq r \leq n$, $-10^7 \leq x \leq 10^7$. Это означает, что ко всем элементам массива a на отрезке от l до r нужно прибавить x .

Запрос третьего типа задается так: $3\ l\ r$

Где $1 \leq l \leq r \leq n$. Это означает, что нужно вывести сумму элементов массива a на отрезке от l до r .

Формат выходных данных

Для каждого запроса третьего типа выведите в отдельной строке сумму элементов на соответствующем отрезке.

Примеры

стандартный ввод	стандартный вывод
3 1 4 2 9 3 1 3 1 1 3 3 3 1 3 1 1 3 1 3 1 3 2 1 3 5 3 1 3 1 1 3 3 3 1 3	7 6 3 18 9
7 1 7 2 4 8 4 100 10 1 3 6 3 3 2 7 1 2 3 5 2 3 4 -10 3 1 7 1 1 7 3 3 1 4 3 2 7 2 1 7 5 3 1 7	118 97 -11 -3 33

Задача С. Минимизируй, прибавляй, НОДируй и их друзья

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	4 секунды
Ограничение по памяти:	512 мегабайт

Дан массив целых чисел a длины n . Поступает q запросов восьми типов:

- $1\ l\ r\ x$. Для каждого i на отрезке от l до r включительно нужно заменить a_i на $\min(a_i, x)$.
- $2\ l\ r\ x$. Для каждого i на отрезке от l до r включительно нужно заменить a_i на $\max(a_i, x)$.
- $3\ l\ r\ x$. Для каждого i на отрезке от l до r включительно нужно заменить a_i на x .
- $4\ l\ r\ x$. Для каждого i на отрезке от l до r включительно нужно заменить a_i на $a_i + x$.
- $5\ l\ r$. Необходимо вывести сумму элементов массива a на отрезке от l до r включительно.
- $6\ l\ r$. Необходимо вывести минимум элементов массива a на отрезке от l до r включительно.
- $7\ l\ r$. Необходимо вывести максимум элементов массива a на отрезке от l до r включительно.
- $8\ l\ r$. Необходимо вывести НОД (наибольший общий делитель) элементов массива a на отрезке от l до r включительно.

Формат входных данных

В первой строке входных данных дано целое число n ($1 \leq n \leq 200\,000$) — количество элементов массива a .

Во второй строке даны n целых чисел a_i ($1 \leq a_i \leq 10^9$) — элементы массива a .

В третьей строке дано целое число q ($1 \leq q \leq 200\,000$) — количество запросов.

В последующих q строках даны запросы по одному в строке.

Запрос первого типа задается так: $1\ l\ r\ x$

Где $1 \leq l \leq r \leq n$, $1 \leq x \leq 10^9$ — целые числа. Это означает, что все элементы массива a на отрезке от l до r нужно заменить на минимум из текущего значения и x .

Запрос второго типа задается так: $2\ l\ r\ x$

Где $1 \leq l \leq r \leq n$, $1 \leq x \leq 10^9$ — целые числа. Это означает, что все элементы массива a на отрезке от l до r нужно заменить на максимум из текущего значения и x .

Запрос третьего типа задается так: $3\ l\ r\ x$

Где $1 \leq l \leq r \leq n$, $1 \leq x \leq 10^9$ — целые числа. Это означает, что все элементы массива a на отрезке от l до r нужно заменить на x .

Запрос четвертого типа задается так: $4\ l\ r\ x$

Где $1 \leq l \leq r \leq n$, $1 \leq x \leq 10^7$ — целые числа. Это означает, что ко всем элементам массива a на отрезке от l до r нужно прибавить x .

Запрос пятого типа задается так: $5\ l\ r$

Где $1 \leq l \leq r \leq n$ — целые числа. Это означает, что нужно вывести сумму элементов массива a на отрезке от l до r .

Запрос шестого типа задается так: $6\ l\ r$

Где $1 \leq l \leq r \leq n$ — целые числа. Это означает, что нужно вывести минимум элементов массива a на отрезке от l до r .

Запрос седьмого типа задается так: $7\ l\ r$

Где $1 \leq l \leq r \leq n$ — целые числа. Это означает, что нужно вывести максимум элементов массива a на отрезке от l до r .

Запрос восьмого типа задается так: $8\ l\ r$

Где $1 \leq l \leq r \leq n$ — целые числа. Это означает, что нужно вывести наибольший общий делитель элементов массива a на отрезке от l до r .

Формат выходных данных

Для каждого запроса 5, 6, 7 и 8 типов выведите в отдельной строке ответ.

Пример

стандартный ввод	стандартный вывод
7	71
1 2 3 4 5 6 7	1
20	16
4 2 7 10	1
5 1 6	90
6 1 6	14
7 1 6	17
8 1 6	1
2 1 6 14	74
5 2 7	12
6 2 7	14
7 2 7	2
8 2 7	101
1 2 7 12	12
5 1 6	15
6 1 6	1
7 1 6	
8 1 6	
3 2 6 15	
5 1 7	
6 1 7	
7 1 7	
8 1 7	

Задача D. Алгоритм двух китайцев

Имя входного файла: `chinese.in`
Имя выходного файла: `chinese.out`
Ограничение по времени: 6 секунд
Ограничение по памяти: 256 мегабайт

Вам дан взвешенный ориентированный граф, содержащий n вершин и m рёбер. Найдите минимально возможную сумму весов $n - 1$ ребра, которые нужно оставить в графе, чтобы из вершины с номером 1 по этим ребрам можно было добраться до любой другой вершины.

Формат входных данных

В первой строке даны два целых числа n и m ($1 \leq n \leq 1000$, $0 \leq m \leq 10000$) — количество вершин и ребер в графе.

В следующих m строках даны ребра графа. Ребро описывается тройкой чисел a_i , b_i и w_i ($1 \leq a_i, b_i \leq n$; $-10^9 \leq w_i \leq 10^9$) — номер вершины, из которой исходит ребро, номер вершины, в которую входит ребро, и вес ребра.

Формат выходных данных

Если нельзя оставить подмножество ребер так, чтобы из вершины с номером 1 можно было добраться до любой другой, в единственной строке выведите «NO».

Иначе, в первой строке выведите «YES», а во второй строке выведите минимальную возможную сумму весов ребер, которых необходимо оставить.

Примеры

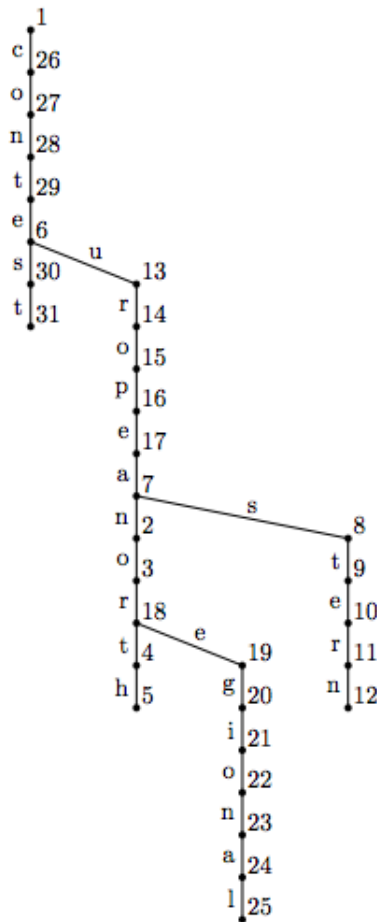
<code>chinese.in</code>	<code>chinese.out</code>
2 1 2 1 10	NO
4 5 1 2 2 1 3 3 1 4 3 2 3 2 2 4 2	YES 6

Задача Е. Словарь

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

Петя и Дима работают над новым алгоритмом сжатия данных. Их задача состоит в том, что сжать данный набор слов. Для этого они хотят построить корневое дерево, где на каждом ребре написана ровно одна буква.

Определим для такого дерева словарь, который содержит в точности те слова, которые могут быть получены конкатенацией букв на некотором пути в этом дереве (не обязательно начинающемся в корне), идущим вниз к листу (но не обязательно заканчивающимся в листе).



Ребята хотят построить такое дерево, для которого соответствующий словарь будет содержать все слова из исходного множества (и, возможно, какие-то еще слова). Среди таких деревьев они хотят выбрать то, которое содержит минимальное количество вершин. Помогите им!

На картинке выше корень дерева имеет номер 1, а путь от вершины 7 до вершины 5 соответствует слову «north», путь от вершины 16 до вершины 12 соответствует слову «eastern», путь от вершины 29 до вершины 2 соответствует слову «european», путь от вершины 3 до вершины 25 соответствует слову «regional», а путь от вершины 1 до вершины 31 соответствует слову "contest".

Формат входных данных

Первая строка входных данных содержит число слов в множестве n ($1 \leq n \leq 50$). Следующие n строк содержат различные непустые слова, по одному на строке, каждое из которых состоит из маленьких английских букв. Каждое слово состоит из не более, чем 10 символов.

Формат выходных данных

В первой строке выведите количество вершин в исходном дереве m . В следующих m строках выведите описания вершин дерева. Вершины нумеруются с 1, описание вершины состоит из номера вершины-предка и символа, написанного на ребре, ведущего предка. Для корневой вершины описание должно состоять из единственного числа 0.

Пример

стандартный ввод	стандартный вывод
5	31
north	0
eastern	1 c
european	2 o
regional	3 n
contest	4 o
	5 r
	6 e
	7 u
	8 r
	9 o
	10 p
	11 e
	12 a
	13 n
	13 s
	15 t
	16 e
	17 r
	18 n
	7 g
	20 i
	21 o
	22 n
	6 t
	24 h
	23 a
	26 l
	4 t
	28 e
	29 s
	30 t

Замечание

Пример соответствует рисунку из условия.

Задача F. Алгоритм двух китайцев. Эпизод второй

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Вам дан взвешенный ориентированный граф, содержащий n вершин и m рёбер. Найдите минимально возможную сумму весов $n - 1$ ребра, которые нужно оставить в графе, чтобы из вершины с номером 1 по этим ребрам можно было добраться до любой другой вершины.

Формат входных данных

В первой строке даны два целых числа n и m ($1 \leq n \leq 300\,000$, $0 \leq m \leq 300\,000$) — количество вершин и ребер в графе.

В следующих m строках даны ребра графа. Ребро описывается тройкой чисел a_i , b_i и w_i ($1 \leq a_i, b_i \leq n$; $-10^9 \leq w_i \leq 10^9$) — номер вершины, из которой исходит ребро, номер вершины, в которую входит ребро, и вес ребра.

Формат выходных данных

Если нельзя оставить подмножество ребер так, чтобы из вершины с номером 1 можно было добраться до любой другой, в единственной строке выведите «NO».

Иначе, в первой строке выведите «YES», а во второй строке выведите минимальную возможную сумму весов ребер, которых необходимо оставить.

Примеры

стандартный ввод	стандартный вывод
2 1 2 1 10	NO
4 5 1 2 2 1 3 3 1 4 3 2 3 2 2 4 2	YES 6

Задача G. Два пути

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

В Байтландии города соединены односторонними дорогами. Дорожная система Байтландии обладает одним интересным свойством: если вы выехали из города по какой-то дороге, вы не сможете вернуться назад. Иными словами, структура дорог может быть описана ориентированным ациклическим графом.

Такая особенность создает некоторые проблемы. Например, до некоторых городов нельзя добраться даже из столицы Байтландии. Бывает и хуже: дороги часто закрываются на ремонт. В таком случае до некоторых городов нельзя будет добраться, даже если до них можно было доехать ранее.

Байтазер живет в столице и часто путешествует в другие города Байтландии. Для каждого города C он хотел бы знать, существуют ли два пути из столицы в C , которые не имеют общих дорог. Если это правда (или C является столицей), Байтазер знает, что путешествие до C всегда возможно, даже если какая-то из дорог закрыта на ремонт. Помогите Байтазеру найти все города, до которых можно добраться из столицы, даже если какая-нибудь одна дорога закроется на ремонт.

Формат входных данных

Первая строка содержит два числа n, m ($1 \leq n \leq 200\,000$, $1 \leq m \leq 500\,000$) – количество городов и дорог в Байтландии. Города пронумерованы числами $1, 2, \dots, n$, столица имеет номер 1. Следующие m строк описывают дорог: i -я из них содержит два числа u_i, v_i ($1 \leq u_i, v_i \leq n$, $u_i \neq v_i$) означающие, что i -я односторонняя дорога начинается в городе u_i и заканчивается в городе v_i .

Формат выходных данных

В первой строке выведите количество городов, до которых можно добраться до столицы, даже если одна из дорог закроется на ремонт. В следующей строке выведите номера этих городов по возрастанию, разделенных пробелами.

Пример

стандартный ввод	стандартный вывод
7 9	4
1 2	1 4 5 7
1 3	
3 4	
4 5	
2 4	
2 5	
5 6	
5 7	
5 7	

Задача Н. Полезные дороги

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	4 секунды
Ограничение по памяти:	512 мегабайт

В столице Берляндии n перекрестков и m дорог, соединяющих их. По дорогам можно передвигаться лишь в одном направлении. Как вы догадываетесь, в Берляндии две беды: дураки и дороги.

До выборов мэра остался один месяц, поэтому сейчас самое время для текущего правительства делать город лучше. Чтобы показать, что они заботятся об инфраструктуре и бюджете, правительство решило починить только *полезные* дороги.

Текущий мэр считает дорогу от перекрестка u к перекрестку v полезной, если существует простой путь, содержащий дорогу (u, v) , начинающийся в мэрии и заканчивающийся в каком-либо перекрестке. Путь называется простым, если никакой перекресток на этом пути не повторяется. Мэрия находится на перекрестке номер 1.

Помогите министерству транспорта найти все полезные дороги в городе.

Формат входных данных

Вход содержит несколько тестовых примеров. Каждый тест начинается со строки, содержащей два целых числа n и m ($2 \leq n \leq 2 \cdot 10^5; 1 \leq m \leq 2 \cdot 10^5$) — количество перекрестков и дорог в городе. Следующие m строк описывают дороги. Описание состоит из пары чисел u_i и v_i ($1 \leq u_i, v_i \leq n; u_i \neq v_i$), означающих, что i -я дорога начинается в перекрестке u_i и заканчивается в перекрестке v_i . Перекрестки пронумерованы от 1 до n . Мэрия находится на перекрестке номер 1.

Гарантированно, что между каждой парой перекрестков не более одной дороги в каждом направлении.

Каждый тестовый пример завершается пустой строкой. Сумма n по всем тестовым примерам не превосходит $2 \cdot 10^5$. Аналогично для m : сумма m по всем тестовым примерам не превосходит $2 \cdot 10^5$.

Формат выходных данных

Для каждого тестового примера выведите две строки. В первой строке должно находиться количество полезных дорог. Во второй строке находятся индексы полезных дорог в возрастающем порядке. Дороги пронумерованы от 1 до m в том же порядке, как они идут во входном файле. Если полезных дорог нет, оставьте вторую строку пустой.

Пример

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

Задача I. Исторический максимум

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 3 секунды
Ограничение по памяти: 512 мегабайт

Дан массив a . Поступают запросы прибавления на отрезке и поиска максимума исторических максимумов на отрезке. Необходимо их обрабатывать.

Формат входных данных

В первой строке дано число n ($1 \leq n \leq 300\,000$) — количество элементов массива a .

Во второй строке даны n чисел — элементы массива a ($-10^9 \leq a_i \leq 10^9$).

В третьей строке дано число q ($1 \leq q \leq 300\,000$) — количество запросов.

В последующих q строках даны запросы.

Запрос прибавления на отрезке задается следующим образом: $1\ ql\ qr\ x$

Это означает, что на отрезке от ql до qr включительно ($1 \leq ql \leq qr \leq n$) нужно прибавить ко всем числам x ($-10^9 \leq x \leq 10^9$).

Запрос поиска исторического максимума задается следующим образом: $2\ ql\ qr$

Это означает, что нужно найти максимум исторических максимумов на отрезке от ql до qr включительно ($1 \leq ql \leq qr \leq n$).

Формат выходных данных

Для каждого запроса второго типа выведите ответ в отдельной строке.

Пример

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