

## Задача А. Это база, это сдать надо!

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

Задан массив  $a_1, \dots, a_n$ , поступают  $q$  запросов, которые имеют следующий вид:

- 1 i x. Присвоить  $a_i$  значение  $x$ ;
- 2 l r x. Для всех  $l \leq i \leq r$  присвоить  $a_i$  значение  $a_i \bmod x$ ;
- 3 l r. Вычислить сумму  $a_l + \dots + a_r$ .

### Формат входных данных

В первой строке задано число  $n$  ( $1 \leq n \leq 100\,000$ ) — длина массива  $a$ .

Во второй строке указаны числа  $a_1, \dots, a_n$  ( $0 \leq a_i \leq 10^9$ ).

В третьей строке задано число  $q$  ( $1 \leq q \leq 100\,000$ ) — количество запросов.

В последующих строках находятся запросы, на которые действуют следующие ограничения:

- 1 i x ( $1 \leq i \leq n, 0 \leq x \leq 10^9$ )
- 2 l r x ( $1 \leq l \leq r \leq n, 1 \leq x \leq 10^9$ )
- 3 l r. ( $1 \leq l \leq r \leq n$ )

### Формат выходных данных

Выведите ответы на запросы третьего вида.

### Пример

стандартный ввод	стандартный вывод
6	15
1 2 3 4 5 6	
3	
2 1 5 3	
1 3 3	
3 1 6	

## Задача В. Минимизируй!

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 2 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 С. Кинетическое дерево отрезков

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

Дан массив из  $n$  прямых  $a_i x + b_i$  и  $q$  запросов:

- $1 \ i \ a \ b$  ( $1 \leq i \leq n, 1 \leq a, b \leq 10^9$ ) —  $i$ -я прямая становится равной  $ax + b$
- $2 \ l \ r \ t$  ( $1 \leq l \leq r \leq n, 0 \leq t \leq 10^9$ ) — найти минимум среди прямых на отрезке от  $l$  до  $r$  включительно в точке  $t$

Гарантируется, что значения  $t_i$  не убывают.

### Формат входных данных

Первая строка содержит два целых числа  $n$  и  $q$  ( $1 \leq n, q \leq 10^5$ ) — количество прямых и запросов соответственно.

Вторая строка содержит  $n$  целых чисел  $a_i$  ( $1 \leq a_i \leq 10^9$ ) — изначальные угловые коэффициенты прямых.

Третья строка содержит  $n$  целых чисел  $b_i$  ( $1 \leq b_i \leq 10^9$ ) — изначальные свободные члены прямых.

В следующих  $q$  строках заданы запросы в формате, описанном в условии.

### Формат выходных данных

Выведите ответы на запросы второго типа.

### Пример

стандартный ввод	стандартный вывод
6 10	11
7 6 5 2 6 2	11
10 6 4 7 9 5	15
2 1 4 2	23
1 1 2 3	21
2 1 4 4	87
2 1 6 6	
1 2 8 7	
2 3 5 8	
2 5 6 8	
1 5 10 7	
2 2 2 10	
1 5 9 4	

## Задача D. Сортировки на отрезках

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

Дана перестановка длины  $N$ . Поступает два вида запросов:

- 1 1  $r$ . Требуется отсортировать по возрастанию подотрезок перестановки с  $l$  по  $r$ ;
- 2 1  $r$ . Требуется отсортировать по убыванию подотрезок перестановки с  $l$  по  $r$ .

Выведите перестановку после всех операций.

### Формат входных данных

В первой строке вводятся числа  $N$  и  $Q$  — размер перестановки и число запросов ( $1 \leq N, Q \leq 3 \cdot 10^5$ ).

В следующей строке вводятся  $N$  чисел  $p_1 \dots p_N$  — перестановка чисел от 1 до  $N$ .

В каждой из следующих  $Q$  строк вводятся по три числа  $t, l, r$  — параметры очередного запроса ( $1 \leq t \leq 2, 1 \leq l \leq r \leq N$ ).

### Формат выходных данных

Выведите перестановку после всех операций.

### Примеры

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

## Задача E. Дели, прибавляй!

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 3 секунды  
Ограничение по памяти: 256 мегабайт

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

- $1\ l\ r\ x$ . Для каждого  $i$  на отрезке от  $l$  до  $r$  включительно нужно заменить  $a_i$  на  $a_i + x$ .
- $2\ l\ r\ x$ . Для каждого  $i$  на отрезке от  $l$  до  $r$  включительно нужно заменить  $a_i$  на  $\lfloor \frac{a_i}{x} \rfloor$  ( $\lfloor \cdot \rfloor$  — это округление вниз).
- $3\ l\ r$ . Необходимо вывести минимум элементов массива  $a$  на отрезке от  $l$  до  $r$  включительно.
- $4\ l\ r$ . Необходимо вывести сумму элементов массива  $a$  на отрезке от  $l$  до  $r$  включительно.

### Формат входных данных

В первой строке входных данных даны два целых числа  $n$  и  $q$  ( $1 \leq n, q \leq 200\,000$ ) — количество элементов массива  $a$  и количество запросов.

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

В последующих  $q$  строках даны запросы по одному в строке.

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

Где  $0 \leq l \leq r < n$ ,  $-10^7 \leq x \leq 10^7$  — целые числа. Это означает, что ко всем элементам массива  $a$  на отрезке от  $l$  до  $r$  нужно прибавить  $x$ .

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

Где  $0 \leq l \leq r < n$ ,  $1 \leq x \leq 10^9$  — целые числа. Это означает, что все элементы массива  $a$  на отрезке от  $l$  до  $r$  нужно поделить на  $x$  и округлить вниз.

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

Где  $0 \leq l \leq r < n$  — целые числа. Это означает, что нужно вывести минимум элементов массива  $a$  на отрезке от  $l$  до  $r$ .

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

Где  $0 \leq l \leq r < n$  — целые числа. Это означает, что нужно вывести сумму элементов массива  $a$  на отрезке от  $l$  до  $r$ .

### Формат выходных данных

Для каждого запроса 3 и 4 типов выведите в отдельной строке ответ.

### Пример

стандартный ввод	стандартный вывод
7 7	-10
10 -6 4 7 12 1 0	-10
2 1 4 4	-34
1 0 5 -8	-36
3 0 6	-26
3 1 5	
4 0 6	
4 1 5	
4 2 6	

### Замечание

Обратите внимание на то, что элементы массива нумеруются с нуля.

## Задача F. 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 G. Исторический максимум

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

Задан массив  $a_1, \dots, a_n$ . Поступают запросы изменения массива. После каждого запроса можно определить массив  $b_1, \dots, b_n$ , где  $b_i$  — максимальное значение, которое принимала переменная  $a_i$  с исходного по текущий момент.

Запросы изменения бывают двух видов:

- 1 1 r x. Требуется прибавить  $x$  ко всем  $a_i$  ( $l \leq i \leq r$ );
- 2 1 r. Требуется вычислить сумму  $b_l + b_{l+1} + \dots + b_r$ .

### 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.

- 1 1 r x ( $1 \leq l \leq r \leq n$ ,  $-10^9 \leq x \leq 10^9$ );
- 2 1 r ( $1 \leq l \leq r \leq n$ ).

### Формат выходных данных

Для каждого запроса второго типа выведите ответ в отдельной строке.

### Пример

стандартный ввод	стандартный вывод
5	20
1 2 3 4 5	11
5	23
1 1 4 2	
2 2 5	
1 3 5 -5	
2 4 5	
2 1 5	



## Задача Н. Выкапываем кости динозавров

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	5 секунд
Ограничение по памяти:	512 мегабайт

Палеонтологи ищут кости динозавров! Они уже нашли длинную линию из  $n$  квадратных секторов и пронумеровали их целыми числами от 1 до  $n$ . Длина стороны каждого квадрата составляет 1 метр. Предварительные измерения показали, что в секторе  $i$  глубина почвы, потенциально содержащей кости динозавров, составляет  $a_i$  метров. Ниже этой глубины находится твердый грунт. Все числа  $a_i$  оказались различными целыми числами.

Ученые подготовили  $q$  различных планов для своих исследований. Каждый план включает в себя строительство исследовательской станции на подотрезке секторов, пронумерованных от  $\ell_j$  до  $r_j$ . После выбора подотрезка они выберут один из его секторов  $m$  ( $\ell_j \leq m \leq r_j$ ) в качестве основного сектора.

Специальное устройство будет закопано в основном секторе на глубине  $a_m$  метров. Это устройство позволяет исследователям анализировать верхние  $a_m$  метров всех секторов под исследовательской станцией, которые имеют глубину **строго больше**  $a_m$ . В общей сложности будет проанализировано  $a_m \cdot k$  кубических метров почвы, где  $k$  — это количество секторов под станцией (то есть между  $\ell_j$  и  $r_j$ , включая их), которые глубже основного сектора.

Палеонтологи хотят найти как можно больше костей динозавров, поэтому они хотят проанализировать как можно больше почвы. Помогите им! Найдите максимальный объем почвы, который можно проанализировать, если подотрезок выбран из планов, и его основной сектор затем выбран оптимально.

### Формат входных данных

Первая строка содержит одно целое число  $n$ , количество секторов ( $1 \leq n \leq 10^6$ ).

Вторая строка содержит  $n$  различных целых чисел  $a_1, a_2, \dots, a_n$  — глубины секторов ( $1 \leq a_i \leq 10^9$ ).

Следующая строка содержит одно целое число  $q$  — количество планов ( $1 \leq q \leq 10^6$ ).

Каждая из следующих  $q$  строк описывает план.  $j$ -я из них содержит два целых числа  $\ell_j$  и  $r_j$ , которые являются концами подотрезка для  $j$ -го плана ( $1 \leq \ell_j \leq r_j \leq n$ ).

### Формат выходных данных

Выведите одно целое число: максимальный объем проанализированной почвы в кубических метрах.

### Пример

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

### Замечание

В примере ученые должны выбрать первый план и первый сектор в качестве основного сектора. Тогда будет проанализировано  $3 \cdot 3 = 9$  (так как глубины 5, 7, 4 больше 3) кубических метров почвы.

## Задача I. Битовые операции

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 3 секунды  
Ограничение по памяти: 256 мегабайт

Дан массив  $a_1, a_2, \dots, a_n$ , состоящий из целых неотрицательных чисел. Вам требуется последовательно совершить  $m$  операций, которые могут быть трех видов:

- 1 1 r x. Присвоить  $a_i := a_i \text{ and } x$  для всех  $l \leq i \leq r$ ;
- 2 1 r x. Присвоить  $a_i := a_i \text{ or } x$  для всех  $l \leq i \leq r$ ;
- 3 1 r. Вычислить  $\min\{a_l, a_{l+1}, \dots, a_r\}$ .

Операторы `and` и `or` означают «*битовое и*» и «*битовое или*», соответственно.

### Формат входных данных

В первой строке указана пара чисел  $n$  и  $m$  ( $n, m \leq 500\,000$ ) — длина массива  $a$  и количество запросов.

Во второй строке указаны числа  $a_1, \dots, a_n$  ( $0 \leq a_i < 2^{31}$ ).

В последующих  $m$  строках указаны запросы. Гарантируется  $1 \leq l \leq r \leq n$  и  $0 \leq x < 2^{31}$ .

### Формат выходных данных

Последовательно выведите результаты ответов на запросы третьего вида.

### Пример

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

## Задача J. Продвинутое кинетическое дерево отрезков

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

Даны два массива из целых чисел  $a_i$  и  $b_i$  длины  $n$ . Поступают  $q$  запросов:

- $1\ l\ r\ x$  ( $0 \leq x \leq 10^6, 1 \leq l \leq r \leq n$ ) — выполнить прибавление  $b_i += a_i \cdot x$  для всех  $l \leq i \leq r$
- $2\ l\ r$  ( $1 \leq l \leq r \leq n$ ) — найти минимум среди значений  $b_i$  на отрезке от  $l$  до  $r$  включительно

### Формат входных данных

Первая строка содержит два целых числа  $n$  и  $q$  ( $1 \leq n, q \leq 10^5$ ) — количество прямых и запросов соответственно.

Вторая строка содержит  $n$  целых чисел  $a_i$  ( $1 \leq a_i \leq 10^6$ ).

Третья строка содержит  $n$  целых чисел  $b_i$  ( $1 \leq b_i \leq 10^6$ ).

В следующих  $q$  строках заданы запросы в формате, описанном в условии.

### Формат выходных данных

Выведите ответы на все запросы второго типа.

### Пример

стандартный ввод	стандартный вывод
6 10	1
3 6 7 5 7 9	29
6 5 1 9 10 7	7
2 3 5	9
1 3 3 4	9
2 3 3	9
1 1 3 2	7
2 4 6	
2 2 4	
2 4 4	
1 5 5 5	
2 2 5	
2 6 6	

## Задача К. Интересные вершины

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

Дано корневое дерево на  $n$  вершинах, пронумерованных числами от 1 до  $n$ . Корень находится в вершине 1.

У каждой вершины есть пара параметров  $a_i$  и  $b_i$ . Обозначим множество всех предков  $v$  (включая и саму вершину  $v$ ) как  $R(v)$ . Обозначим *интересность* вершины  $v$  как:

$$\left| \sum_{u \in R(v)} a_u \right| \cdot \left| \sum_{u \in R(v)} b_u \right|$$

Нужно выполнить  $q$  запросов следующего вида:

- 1  $v$   $x$ . Увеличить  $a_v$  на положительное число  $x$ ;
- 2  $v$ . Найти максимальное значение *интересности* в поддереве  $v$ .

### Формат входных данных

В первой строке заданы числа  $n$  и  $q$  ( $1 \leq n \leq 2 \cdot 10^5$ ,  $1 \leq q \leq 10^5$ ) — количество вершин в поддереве и количество запросов соответственно.

Вторая строка содержит числа  $p_2, \dots, p_n$  ( $1 \leq p_i < i$ ) — номера предков у вершин  $2, 3, \dots, n$ .

Третья строка содержит числа  $a_1, \dots, a_n$  ( $-5\,000 \leq a_i \leq 5\,000$ ).

Четвертая строка содержит числа  $b_1, \dots, b_n$  ( $-5\,000 \leq b_i \leq 5\,000$ ).

Каждая из последующих  $q$  строк описывает запрос и имеет вид:

- 1  $v$   $x$  ( $1 \leq v \leq n$ ,  $1 \leq x \leq 5\,000$ );
- 2  $v$  ( $1 \leq v \leq n$ )

### Формат выходных данных

Для каждого запроса второго типа в отдельной строке выведите максимальное значение *интересности* вершины в соответствующем поддереве.

### Пример

стандартный ввод	стандартный вывод
5 6	100
1 1 2 2	91
10 -3 -7 -3 -10	169
10 3 9 3 6	240
2 1	
2 2	
1 2 6	
2 1	
1 2 5	
2 1	

## Задача L. The Third Grace

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

Даны  $n$  отрезков и  $m$  точек на числовой прямой.  $i$ -й отрезок имеет границы  $[l_i, r_i]$ , а  $j$ -я точка имеет координату  $j$  и имеет коэффициент  $p_j$ .

Вы выбираете некоторое подмножество указанных  $m$  точек. Для каждого из  $n$  отрезков его стоимость определяется как:

- 0, если отрезок не содержит точек из подмножества;
- коэффициент точки из подмножества с **наибольшей координатой** внутри отрезка в противном случае.

Ваша задача — найти подмножество точек, которое максимизирует сумму стоимостей всех отрезков.

### Формат входных данных

Каждый тест содержит несколько тестовых данных. Первая строка содержит одно целое число  $t$  ( $1 \leq t \leq 10^5$ ) — количество наборов входных данных. Затем следует описание наборов.

Первая строка каждого набора содержит два целых числа  $n$  и  $m$  ( $1 \leq n, m \leq 10^6$ ) — количество отрезков и количество точек.

В следующих  $n$  строках каждого набора указаны границы отрезков  $l_i$  и  $r_i$  ( $1 \leq l_i \leq r_i \leq m$ ).

Следующая строка каждого набора содержит числа  $p_1, \dots, p_m$  ( $0 \leq p_i \leq 10^9$ ) — коэффициенты точек.

Гарантируется, что сумма  $n$  по всем наборам не превышает  $10^6$ , и сумма  $m$  не превышает  $10^6$ .

### Формат выходных данных

Выведите максимальную возможную суммарную стоимость всех отрезков.

### Пример

стандартный ввод	стандартный вывод
2	108
2 8	0
1 5	
3 8	
78 0 50 0 0 0 0 30	
1 6	
1 5	
0 0 0 0 0 100	

## Задача М. Случайный минимакс

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 512 мегабайт

Дано бинарное дерево. Его вершины пронумерованы от 1 до  $n$ , оно подвешено за вершину 1. У листьев есть веса  $w_i$ . Все веса различные. У каждой вершины, не являющейся листом, задан параметр — вероятность  $p_i$ .

Запускается случайный процесс установки весов вершин (не листьев) от более глубоких к менее глубоким. На очередном шаге вес вершины  $v$  устанавливается как максимум из весов её детей с вероятностью  $p_i$ , или как минимум с вероятностью  $1 - p_i$ .

В итоге будет установлен вес корня. Обозначим все возможные веса корня за  $D_i$  в порядке возрастания. Вероятность того, что вес в корне равен  $D_i$ , обозначим за  $V_i$ . От вас требуется найти значение величины

$$\sum_{i=1}^n i \cdot D_i \cdot V_i^2$$

От вас требуется вывести это число по модулю 998244353. Можно доказать, что ответ представим в виде несократимой дроби  $\frac{p}{q}$ , тогда вам нужно вывести число  $p \cdot q^{-1} \pmod{998244353}$ .

### Формат входных данных

В первой строке входных данных находится одно число  $n$  ( $1 \leq n \leq 3 \cdot 10^5$ ).

В следующей строке вводятся числа  $t_1 \dots t_n$ . Очередное  $t_i$  ( $0 \leq t_i < i$ ) это номер предка  $i$ -й вершины, у корня нет предка, для него вводится 0.

В третьей строке вводятся целые числа  $r_i$ , из них получаются  $p_i$  и  $w_i$  таким образом:  $w_i = r_i$ , если вершина  $i$  — лист, и  $p_i = \frac{r_i}{10000}$ , иначе. Гарантируется, что  $1 \leq w_i \leq 10^9$ ,  $0 < p_i < 1$ .

### Формат выходных данных

Выведите одно число — значение искомой суммы по модулю 998244353.

### Примеры

стандартный ввод	стандартный вывод
3 0 1 1 5000 1 2	748683266
7 0 1 1 2 2 3 3 8000 5000 7500 1 4 3 2	980151178

### Замечание

В первом примере вероятности того, что в корне окажется 1 и 2 одинаковые и равны  $\frac{1}{2}$ , поэтому ответ это  $1 \cdot 1 \cdot (\frac{1}{2})^2 + 2 \cdot 2 \cdot (\frac{1}{2})^2 = \frac{5}{4}$ .

Во втором примере вероятности в корне может оказаться число 1 с вероятностью  $\frac{1}{10}$ , число 2 с вероятностью  $\frac{1}{8}$ , число 3 с вероятностью  $\frac{3}{8}$  и число 4 с вероятностью  $\frac{2}{5}$ .

## Задача N. Принцип Арнольда

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 3 секунды  
Ограничение по памяти: 256 мегабайт

Дан массив  $a_1, \dots, a_n$ , с которым надо уметь развлекаться следующим образом:

- 1 1 r x. Для всех  $l \leq i \leq r$  значения  $a_i$  увеличиваются на  $x$ .
- 2 1 r. Требуется вычислить следующую сумму:

$$a_l + \max\{a_l, a_{l+1}\} + \max\{a_l, a_{l+1}, a_{l+2}\} + \dots + \max\{a_l, \dots, a_r\}$$

### Формат входных данных

В первой строке указано числа  $n$  и  $q$  ( $1 \leq n, q \leq 300\,000$ ) — количество элементов в массиве и количество запросов.

Во второй строке указаны числа  $a_1, \dots, a_n$  ( $|a_i| \leq 10^6$ ) — исходные элементы в массиве.

Затем следуют  $q$  строк, содержащих описание запросов в следующем формате:

- 1 1 r x ( $1 \leq l \leq r \leq n, |x| \leq 10^6$ )
- 2 1 r ( $1 \leq l \leq r \leq n$ )

### Формат выходных данных

Выведите ответы на запросы второго типа, каждый в своей строке.

### Пример

стандартный ввод	стандартный вывод
5 7	15
1 2 3 4 5	21
2 1 5	14
1 1 1 3	11
2 1 5	-7
2 2 5	
1 3 4 -7	
2 2 5	
2 3 4	

## Задача O. Упражнение для мастера спорта

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

Мастер спорта Мг К пришел в спортзал, чтобы познать штангу, но с ним в спортзал зашел и Мг В. Мг В очень разочарован в том, что Мг К тратит время на какую-то ерунду вместо настоящего спорта, а поэтому он решил взять Мг К на слабо, заодно заставив его напрячь голову и решить задачу по информатике.

В спортивном зале изначально в ряд стоит  $n$  пар блинов с массами  $a_1$  и  $b_1$ ,  $a_2$  и  $b_2$ , ...,  $a_n$  и  $b_n$ . Затем происходит  $q$  событий одного из двух типов:

- 1  $i$   $x$   $y$ . Менеджер меняет пару блинов на позиции  $i$  на пару блинов с массами  $(x, y)$ , после чего выполнено  $a_i = x$  и  $b_i = y$ .
- 2 1  $r$ . Мг К покупает абонемент в зал, который позволяет ему выбрать любую пару блинов  $l \leq i \leq j \leq r$ , после чего Мг К жмет штангу весом  $a_i + b_j$ . Но так как в зал также пришел еще и Мг В, то Мг К успевает выбрать только индекс  $i$ , после чего Мг В выбирает любой подходящий индекс  $j$ .

Известно, что Мг В всегда выбирает индекс  $j$  так, чтобы максимизировать суммарный вес штанги, которую будет жать Мг К. Мг К это не очень нравится, так как он начал ходить в зал только недавно, поэтому каждый раз он выбирает индекс  $i$  так, чтобы минимизировать вес штанги  $a_i + b_j$ .

Мг К не просто так является мастером спорта, а поэтому он знает как написать программу, которая выбирает оптимальный индекс  $i$ . К сожалению, сейчас у Мг К нет с собой ноутбука, поэтому он написал вам с просьбой о помощи.

### Формат входных данных

- В первой строке указана пара чисел  $n$  и  $q$  ( $1 \leq n, q \leq 300\,000$ ).  
Во второй строке указаны числа  $a_1, \dots, a_n$  ( $1 \leq a_i \leq 10^9$ ).  
В третьей строке указаны числа  $b_1, \dots, b_n$  ( $1 \leq b_i \leq 10^9$ ).  
В последующих  $q$  строках описаны события в следующем формате:

- 1  $i$   $x$   $y$  ( $1 \leq i \leq n, 1 \leq x, y \leq 10^9$ )
- 2 1  $r$  ( $1 \leq l \leq r \leq n$ )

### Формат выходных данных

Для каждого события 2-го типа выведите минимальный вес штанги, которую будет жать Мг К.

### Пример

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



## Задача Р. Горячие обеды и очереди

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	4 секунды
Ограничение по памяти:	256 мегабайт

Как известно, в системе образования Берляндии есть большая проблема - наглые школьники влезаящие вне очереди в столовой. Чтобы оценить серьезность проблемы в столовую прибыли делегаты из министерства образования. За время своей проверки они смогли понять как работает очередь: У  $i$  человека есть два параметра -  $num_i, val_i$  - номер  $i$  человека в очереди, и его недовольство. Изначально очередь пустая. Туда начинают вставать школьники в порядке увеличения их номеров. Если  $i$  человек встает на место  $j$ , то у всех школьников с номерами не меньшими  $j$ , номер и недовольство увеличивается на 1.

Например, если очередь выглядела так:  $[1] \rightarrow [1, 2] \rightarrow [1, 3, 2] \rightarrow [1, 3, 2, 4] \rightarrow [1, 3, 5, 2, 4]$

То недовольства школьников были следующие:  $[0] \rightarrow [0, 0] \rightarrow [0, 0, 1] \rightarrow [0, 0, 1, 0] \rightarrow [0, 0, 0, 2, 1]$

Теперь вы хотите понять количество людей вставших не в конец очереди, то есть количество школьников  $i$ , что встали не на место  $i$ . Но вот незадача человек, ответственный за подсчет этой величины не смог вытерпеть и встал в очередь. Поэтому вы хотите спросить у каждого школьника его недовольство, и по этой информации узнать ответ, то есть вам нужно понять количество школьников вставших вне очереди по массиву недовольств.

Но министерство образования злое, поэтому оно решило усложнить задачу. Поэтому изначально вам дан массив недовольств очереди, и вам требуется отвечать на два вида запросов: 1: С прибавить некое число  $add$  на отрезке массива недовольств. 2: Сказать какой ответ, если массив недовольств равен  $[val_l, val_{l+1}, \dots, val_r]$ .

Гарантируется, что в любой момент времени  $|val_i| \leq 10^9$ , а также, что очередь корректная.

### Формат входных данных

В первой строке задано единственное число  $n$  ( $3 \leq n \leq 5 \cdot 10^5$ ) - размер массива.

Во второй строке задано  $n$  чисел через пробел  $a_1, a_2, \dots, a_n$  ( $0 \leq a_i < n$ ), где  $a_i$  - недовольство человека под номером  $i$ .

В третьей строке задано единственное число  $q$  ( $0 \leq q \leq 5 \cdot 10^5$ ) - количество запросов на массиве. Далее следует  $q$  строк описание запросов.

Каждая строка начинается с числа  $t$ .

Если  $t = 0$ , то даны еще два числа  $l, r$  ( $1 \leq l \leq r \leq n$ ) - границы запроса. И требуется сказать, какой ответ, если массив недовольств равен  $[val_l, val_{l+1}, \dots, val_r]$ .

Если  $t = 1$ , то даны еще три числа  $l, r, val$  ( $1 \leq l \leq r \leq n$ ), ( $-n < val < n$ ) - границы запроса и значение, что нужно прибавить на отрезке. И требуется для каждого  $i$ , где ( $l \leq i \leq r$ ) выполнить операцию  $a_i = a_i + val$

### Формат выходных данных

В первой строчке вы должны вывести ответ для изначального массива недовольств. Затем для каждого запроса, где  $t = 0$  вывести ответ для него в отдельной строчке.

## Пример

стандартный ввод	стандартный вывод
5	2
0 0 0 2 1	2
6	2
2 4 4 -1	3
2 3 4 1	
1 2 4	
1 2 5	
2 4 5 1	
1 1 5	