Задача A. 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 $a_1, \ldots, a_n$, поступают q запросов, которые имеют следующий вид:

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

Формат входных данных

В первой строке задано число n $(1 \leqslant n \leqslant 100\,000)$ — длина массива a.
Во второй строке указаны числа $a_1, \ldots, a_n$ $(0 \leqslant a_i \leqslant 10^9)$.
В третьей строке задано число q $(1 \leqslant q \leqslant 100\,000)$ — количество запросов.
В последующих строках находятся запросы, на которые действуют следующие ограничения:

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

Формат выходных данных

Выведите ответы на запросы третьего вида.

Пример

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

Задача B. 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 ($1 \leqslant n \leqslant 300\,000$) — количество элементов массива a.

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

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

В последующих q строках даны запросы по одному в строке.

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

Где $1 \leqslant l \leqslant r \leqslant n$, $1 \leqslant x \leqslant 10^9$ — 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 $1 \leqslant l \leqslant r \leqslant n$ — 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 1 4 2 5 2 1 3 1 1 3 3 2 1 3 1 1 3 1 2 1 3	7 6 3
7 1 7 2 4 8 4 100 7 1 3 6 3 2 2 7 1 2 3 5 2 1 7 1 1 7 3 2 1 4 2 2 7	118 117 9 17

Задача C. 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 ($1 \leqslant n \leqslant 300\,000$) — количество элементов массива a.

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

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

В последующих q строках даны запросы по одному в строке.

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

Где $1 \leqslant l \leqslant r \leqslant n$, $-10^9 \leqslant x \leqslant 10^9$. 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 $1 \leqslant l \leqslant r \leqslant n$, $-10^7 \leqslant x \leqslant 10^7$. 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 $1 \leqslant l \leqslant r \leqslant n$. 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 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 l r`. Требуется отсортировать по возрастанию подотрезок перестановки с l по r;
- `2 l 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 ($1 \leqslant N, Q \leqslant 3 \cdot 10^5$).

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

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

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 $\left\lfloor \frac{a_i}{x} \right\rfloor$ ($\lfloor \cdot \rfloor$ — 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 ($1 \leqslant n, q \leqslant 200\,000$) — количество элементов массива a и количество запросов.

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

В последующих q строках даны запросы по одному в строке.

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

Где $0 \leqslant l \leqslant r < n$, $-10^7 \leqslant x \leqslant 10^7$ — 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 $0 \leqslant l \leqslant r < n$, $1 \leqslant x \leqslant 10^9$ — 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 $0 \leqslant l \leqslant r < n$ — 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 $0 \leqslant l \leqslant r < n$ — 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 -6 4 7 12 1 0 2 1 4 4 1 0 5 -8 3 0 6 3 1 5 4 0 6 4 1 5 4 2 6	-10 -10 -34 -36 -26

Замечание

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

Задача F. Исторический максимум

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

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

Запросы изменения бывают двух видов:

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

Формат входных данных

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

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

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

В последующих q строках даны запросы.

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

Формат выходных данных

Для каждого запроса второго типа выведите ответ в отдельной строке.

Пример

<table>
<tr><th>стандартный ввод</th><th>стандартный вывод</th></tr>
<tr><td>5
1 2 3 4 5
5
1 1 4 2
2 2 5
1 3 5 -5
2 4 5
2 1 5</td><td>20
11
23</td></tr>
</table>

Задача G. Выкапываем кости динозавров

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

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 ℓ_j до r_j. После выбора подотрезка они выберут один из его секторов m ($\ell_j \leqslant m \leqslant r_j$) 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 ℓ_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 ($1 \leqslant n \leqslant 10^6$).

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

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

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

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 H. Битовые операции

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

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

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

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

Формат входных данных

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

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

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

Формат выходных данных

Последовательно выведите результаты ответов на запросы третьего вида.

Пример

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

Задача I. Фудкорт

Имя входного файла:	`стандартный ввод`
Имя выходного файла:	`стандартный вывод`
Ограничение по времени:	2.5 секунд
Ограничение по памяти:	512 мегабайт

В огромном торговом центре в Сеуле есть фудкорт специально для больших групп посетителей. На фудкорте расположено N заведений в ряд, пронумерованных от 1 до N. Перед каждым заведением есть отдельная очередь из клиентов.

Сегодня в торговый центр пришли M групп посетителей. Члены группы будут вставать в очереди на фудкорте определённым образом, чтобы им было удобно общаться друг с другом.

Для того чтобы скрасить ожидание посетителей, администрация торгового центра придумала замечательную акцию: время от времени некоторые посетители в очереди будут получать десерт в подарок. Эти десерты разносит Джой, и ей нужна ваша помощь, чтобы определить, кому следует отнести очередной подарочный десерт.

В момент открытия торгового центра, естественно, никто не стоит в очередях на фудкорте. После этого происходят Q событий трёх типов.

- `1 L R C K` — в конец очереди к каждому заведению с номерами между L и R (включительно) встают по K человек из группы C. (Таким образом, в очередях становится суммарно на $K \cdot (R - L + 1)$ людей больше.)
- `2 L R K` — в каждом заведении с номерами между L и R (включительно) освобождается по K мест. Если в очереди к заведению j стоят не больше, чем K посетителей, то все они заходят в это заведение ($L \leqslant j \leqslant R$). Если же в очереди к заведению j стоят больше, чем K посетителей, то первые K из них заходят в заведение ($L \leqslant j \leqslant R$).
- `3 A B` — если в очереди к заведению A находится хотя бы B человек, Джой отдает бесплатный десерт к человеку на позиции B в очереди к заведению A.

Помогите Джой! Определите для каждого десерта, который она должна принести, к какой группе относится счастливый получатель подарка (или то, что в выбранной очереди слишком мало людей, и десерт придется съесть сотрудникам торгового центра).

Формат входных данных

В первой строке входных данных вводятся три целых числа N, M, Q — количество заведений на фудкорте, число групп посетителей в торговом центре и число событий за день, соответственно ($1 \leqslant N, M, Q \leqslant 250\,000$).

В каждой из следующих Q строк вводятся описания событий одного из трёх типов.

- `1 L R C K` — по K людей из группы C встают в конец каждой из очередей перед заведениями с номерами от L до R. Гарантируется, что $1 \leqslant L \leqslant R \leqslant N, 1 \leqslant C \leqslant M, 1 \leqslant K \leqslant 10^9$.
- `2 L R K` — по K посетителей из начала очередей перед заведениями с номерами от L до R проходят в соответствующие заведения (или все люди, стоявшие в очереди, если их было меньше, чем K). Гарантируется, что $1 \leqslant L \leqslant R \leqslant N, 1 \leqslant K \leqslant 10^9$.
- `3 A B` — Джой отдает бесплатный десерт к человеку на позиции B в очереди к заведению A, если в этой очереди есть хотя бы B человек. Гарантируется, что $1 \leqslant A \leqslant N, 1 \leqslant B \leqslant 10^{15}$.

Формат выходных данных

В ответ на каждый запрос третьего типа выведите номер группы C, к которой относится получатель подарочного десерта ($1 \leqslant C \leqslant M$) или число 0, если в очереди не достаточное количество посетителей.

Примеры

стандартный ввод	стандартный вывод
3 5 7 1 2 3 5 2 1 1 2 2 4 3 2 3 2 1 3 3 3 1 2 1 2 3 4 2 3 3 2	2 0 4
3 4 7 1 1 2 1 1 1 1 3 4 1 2 2 3 1 2 1 3 1 1 1 2 2 1 3 1 1 3 3 2	4 0

Замечание

Опишем подробнее первый пример. Будем обозначать очередь скобками и записывать номера групп в людей от начала очереди к концу очереди слева направо.

1. Запрос `1 2 3 5 2` — в очереди 2 и 3 встают по два человека из группы 5. После этого очереди это $(), (5, 5), (5, 5)$.
2. Запрос `1 1 2 2 4` — в очереди 1 и 2 встают по 4 человека из группы 2. После этого очереди это $(2, 2, 2, 2), (5, 5, 2, 2, 2, 2), (5, 5)$.
3. Запрос `3 2 3` — третий человек из очереди 2 получает бесплатный десерт, это человек из группы 2.
4. Запрос `2 1 3 3` — по 3 человека уходят из начала всех очередей (из очереди 3 на самом деле уходят все 2 человека, которые там находились). После этого очереди это $(2), (2, 2, 2), ()$.
5. Запрос `3 1 2` — из очереди 1 десерт должен получить второй человек, но, поскольку в ней ровно один человек, десерт никто не получает.
6. Запрос `1 2 3 4 2` — по 2 человека встают в очереди 2 и 3 из группы 4. После этого очереди это $(2), (2, 2, 2, 4, 4), (4, 4)$.
7. Запрос `3 3 2` — из очереди 3 десерт получает второй человек, он из группы 4.

Задача J. Чистка сугроба

Имя входного файла:	`стандартный ввод`
Имя выходного файла:	`стандартный вывод`
Ограничение по времени:	2 секунды
Ограничение по памяти:	1024 мегабайта

Мальчик Вася очень любит путешествовать. В частности, полёты на самолётах доставляют ему необычайное удовольствие. Он вот-вот должен был вылетать в другой город, но взлётную полосу сильно замело снегом, и она нуждается в очистке.

Взлётную полосу можно представить как n последовательных участков, пронумерованных от 1 до n. Метель была достаточно сильной, но уже прекратилась, поэтому Васе удалось вычислить, что i-й участок завален a_i метрами снега. Для подобных ситуаций в аэропорте есть чистильщик, который работает достаточно необычным образом. За одну минуту чистильщик может сделать следующее:

- Выбрать последовательный отрезок участков длины не более d и убрать по метру снега с самых заваленных участков.

  Формально говоря, можно выбрать $1 \leqslant l \leqslant r \leqslant n$ $(r - l + 1 \leqslant d)$. После этого вычисляется $c = \max\{a_l, a_{l+1}, \ldots, a_r\}$, и если $c > 0$, то для всех $i\colon l \leqslant i \leqslant r$ таких, что $a_i = c$, значение a_i уменьшается на единицу.

Вася долго готовился к полёту и хочет понять, сколько времени ему осталось ждать до полной очистки всех участков от снега. Иначе говоря, требуется вычислить минимальное количество минут, которое придётся потратить чистильщику, чтобы добиться $a_i = 0$ для всех i от 1 до n.

Формат входных данных

Каждый тест состоит из нескольких наборов входных данных. В первой строке находится одно целое число t $(1 \leqslant t \leqslant 2 \cdot 10^5)$ — количество наборов входных данных. Далее следует описание наборов входных данных.

Первая строка каждого набора входных данных содержит два целых числа n и d $(1 \leqslant n \leqslant 5 \cdot 10^5, 1 \leqslant d \leqslant n)$ — количество участков на взлётной полосе и максимальную длину участка, который может выбрать чистильщик.

Вторая строка каждого набора входных данных содержит n целых чисел $a_1, a_2, \ldots, a_n$ $(1 \leqslant a_i \leqslant 10^9)$, где a_i равно количеству метров снега на i участке.

Гарантируется, что сумма n по всем наборам входных данных не превосходит $5 \cdot 10^5$.

Формат выходных данных

Для каждого набора входных данных выведите одно целое число — минимальное количество минут, необходимых чистильщику, чтобы добиться $a_i = 0$ для всех i от 1 до n.

Пример

стандартный ввод	стандартный вывод
2 5 2 1 5 2 1 2 3 1 1000000000 1000000000 1000000000	8 3000000000

Замечание

В первом наборе входных данных существует такая оптимальная последовательность операций. Сначала четыре раза выбрать отрезок $[2, 3]$. После трёх операций a_2 уменьшится до 2, и массив a будет иметь вид $[1, 2, 2, 1, 2]$. После четвёртой операции массив a будет иметь вид $[1, 1, 1, 1, 2]$. Далее можно превратить массив в нули, выбирая отрезки $[1, 2]$, $[3, 3]$, $[5, 5]$ и $[4, 5]$ (именно в таком порядке).

Во втором наборе входных данных $d = 1$, а это значит, что каждый участок очищается независимо от других, и ответ равен сумме всех a_i.

Задача K. Формальное условие

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

Дан массив целых чисел $a_1, a_2, \ldots, a_n$ и целое число k. А также q запросов изменения массива: заменить i-й элемент массива a на x.

После каждого запроса, а также для исходного массива, нужно сказать, чему равно максимальное значение $a_i + a_j$, где $1 \leqslant i < j \leqslant n$, $j - i < k$, а также сколько таких пар (i, j), дающих максимальное значение, существует.

Обратите внимание, что в этой задаче необходимо отвечать на запросы в «онлайне».

Формат входных данных

В первой строке записаны два числа n и k ($2 \leqslant n \leqslant 10^5$, $2 \leqslant k \leqslant n$).

Во второй строке записаны n чисел $a_1, a_2, \ldots, a_n$ ($1 \leqslant a_i \leqslant 10^9$).

В третьей строке записано единственное число q ($1 \leqslant q \leqslant 10^5$). Затем в следующих q строк идет описание запросов изменения.

На каждый запрос вводится два числа i и x ($1 \leqslant i \leqslant n$, $1 \leqslant x \leqslant 10^9$) и пусть ответ для предыдущего запроса $(\mathrm{mx}, \mathrm{cnt})$, где mx — максимальная сумма, а cnt — сколько таких пар было. Тогда нужно будет изменить значение элемента $((i + \mathrm{mx} + \mathrm{cnt}) \bmod n) + 1$ на значение x.

Формат выходных данных

Перед всеми изменениями и после каждого запроса выведите два числа в одной строке — какая максимальная сумма может получиться и сколько таких пар.

Пример

стандартный ввод	стандартный вывод
5 3 5 1 1 1 5 4 1 5 4 2 1 6 3 2	6 4 10 1 7 1 7 3 8 1

Задача L. 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 $p \cdot q^{-1} \mod 998244353$.

Формат входных данных

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

В следующей строке вводятся числа $t_1 ... t_n$. Очередное t_i ($0 \leqslant t_i < i$) 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 $1 \leqslant w_i \leqslant 10^9$, $0 < p_i < 1$.

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 $\frac{2}{5}$.