



Табун лошадей

Мансур любит разводить лошадей, как и его предки. Сейчас у него самый большой табун в Казахстане, но так было далеко не всегда. N лет назад Мансур был простым джигитом (*молодой человек по-казахски*) и у него была всего одна лошадь. Он мечтал заработать много денег и стать баем (*очень богатый человек по-казахски*).

Пронумеруем года числами от 0 до $N - 1$ в хронологическом порядке (то есть так, что номер прошлого года $N - 1$). Каждый год погода влияла на прирост табуна. Для каждого i -го года Мансур помнит целое положительное число $X[i]$ — коэффициент прироста. Если в начале i -го года было h лошадей, то в конце этого же года в табуне становилось $h \cdot X[i]$ лошадей.

Лошадей можно было продавать только в конце года. Для каждого i -го года Мансур помнит целое положительное число $Y[i]$ — цена одной лошади в конце этого года. В конце каждого i -го года можно было продать любое количество лошадей, каждую по цене $Y[i]$.

Мансур хочет узнать максимальное количество денег, которое он смог бы заработать, продавая своих лошадей в самые подходящие моменты в течение N лет. Вы удостоены чести быть гостем на “той” Мансура (*праздник по-казахски*), и он просит ответить на этот вопрос.

Иногда Мансур вспоминает детали, из которых формируется последовательность из M уточнений. Каждое уточнение касается одного из значений $X[i]$ или $Y[i]$. После каждого уточнения он опять просит посчитать максимальное количество денег, которое он мог бы заработать, продавая лошадей. Уточнения применяются последовательно, поэтому каждый ответ на просьбу должен учитывать все предыдущие уточнения. Заметим, что каждое из значений $X[i]$ или $Y[i]$ могут быть уточнены несколько раз.

Ответы на вопросы Мансура могут быть очень большими числами. Чтобы избежать действий с огромными числами, необходимо дать ответ по модулю $10^9 + 7$.

Пример

Предположим, что прошло три года ($N = 3$), и имеется следующая информация:

	0	1	2
X	2	1	3
Y	3	4	1

При таких начальных значениях Мансур заработает максимальное количество денег, продав обе лошади в конце года с номером 1. Последовательность его действий такова:

- Изначально у Мансура одна лошадь.
- В конце года с номером 0 у него будет две лошади ($1 \cdot X[0] = 2$).
- В конце года с номером 1 у него все еще будет две лошади ($2 \cdot X[1] = 2$), сейчас он может их продать, его выгода составит 8 ($2 \cdot Y[1] = 8$).

После этого произошло одно уточнение ($M = 1$), и значение $Y[1]$ стало равным 2.

После уточнения имеем следующую информацию:

	0	1	2
X	2	1	3
Y	3	2	1

В таком случае, одно из оптимальных решений — продать одну лошадь в конце года с номером 0 и три лошади в конце года с номером 2. Последовательность действий такова:

- Изначально у Мансура одна лошадь.
- В конце года с номером 0 у него будет две лошади ($1 \cdot X[0] = 2$), сейчас он может продать одну лошадь, его выгода составит 3 ($1 \cdot Y[0] = 3$). У него останется одна лошадь.
- В конце года с номером 1 у него будет одна лошадь ($1 \cdot X[1] = 1$).
- В конце года с номером 2 у него будет три лошади ($1 \cdot X[2] = 3$), сейчас он может продать три лошади, его выгода составит 3 ($3 \cdot Y[2] = 3$). Таким образом, максимальное суммарное количество денег равно 6 ($3 + 3 = 6$).

Постановка задачи

Даны N , X , Y и список уточнений. Найдите максимальное количество денег (по модулю $10^9 + 7$), которое может заработать Мансур перед всеми уточнениями, а также после каждого уточнения. Вам нужно реализовать функции `init`, `updateX` и `updateY`.

- `init(N, X, Y)` — эта функция вызывается первой и ровно один раз.
 - N — количество прошедших лет.
 - X — массив длины N . $X[i]$ задает коэффициент прироста в конце i -го года ($0 \leq i \leq N - 1$).
 - Y — массив длины N . $Y[i]$ задает цену одной лошади в конце i -го года ($0 \leq i \leq N - 1$).
 - Заметим, что X и Y задают начальные значения, которые задал Мансур (перед всеми уточнениями).
 - После завершения функции `init` область памяти, занятая массивами X и Y , остается корректной. Можно изменять эти массивы, если это необходимо.
 - Функция должна возвращать максимальное количество денег (по модулю $10^9 + 7$), которое Мансур смог бы заработать при начальных значениях X и Y .
- `updateX(pos, val)`
 - `pos` — целое число из отрезка от 0 до $N - 1$.
 - `val` — новое значение для $X[pos]$.
 - Функция должна возвращать максимальное количество денег (по модулю $10^9 + 7$), которое Мансур смог бы заработать с учетом этого уточнения.
- `updateY(pos, val)`
 - `pos` — целое число из отрезка от 0 до $N - 1$.
 - `val` — новое значение для $Y[pos]$.
 - Функция должна возвращать максимальное количество денег (по модулю $10^9 + 7$), которое Мансур смог бы заработать с учетом этого уточнения.

Можно считать, что как начальные, так и уточненные значения массивов $X[i]$ и $Y[i]$, находятся в диапазоне от 1 до 10^9 включительно.

После вызова функции `init` функции `updateX` и `updateY` будут вызваны несколько раз. Суммарное количество вызовов функций `updateX` и `updateY` будет равно M .

Подзадачи

Подзадача	Баллы	N	M	Дополнительные ограничения
1	17	$1 \leq N \leq 10$	$M = 0$	$X[i], Y[i] \leq 10$, $X[0] \cdot X[1] \cdot \dots \cdot X[N-1] \leq 1\,000$
2	17	$1 \leq N \leq 1\,000$	$0 \leq M \leq 1\,000$	—
3	20	$1 \leq N \leq 500\,000$	$0 \leq M \leq 100\,000$	$X[i] \geq 2$ и $val \geq 2$ для функций <code>init</code> и <code>updateX</code> соответственно
4	23	$1 \leq N \leq 500\,000$	$0 \leq M \leq 10\,000$	—
5	23	$1 \leq N \leq 500\,000$	$0 \leq M \leq 100\,000$	—

Пример проверяющего модуля

Проверяющий модуль читает данные из файла `horses.in` в следующем формате:

- Строка 1: N
- Строка 2: $X[0] \dots X[N-1]$
- Строка 3: $Y[0] \dots Y[N-1]$
- Строка 4: M
- Строки 5, ..., $M+4$: три числа `type pos val` (`type=1` для `updateX`, `type=2` для `updateY`).

Проверяющий модуль выводит значение, возвращаемое функцией `init`, затем возвращаемые значения для каждого вызова функции `updateX` и `updateY`.