



Horses

Mansur thích nuôi ngựa, giống như tổ tiên của anh. Anh hiện có đàn ngựa đông nhất ở Kazakhstan. Nhưng điều này không phải lúc nào cũng đúng. Cách đây N năm, Mansur mới chỉ là một dzhigit (tiếng Kazakhstan có nghĩa là *chàng trai trẻ*) và anh chỉ có một con ngựa. Anh đã ước mơ kiếm được thật nhiều tiền và để trở thành một bai (tiếng Kazakhstan có nghĩa là *người rất giàu có*).

Ta đánh số các năm từ 0 đến $N - 1$ theo thứ tự thời gian (nghĩa là năm $N - 1$ là năm gần đây nhất). Thời tiết mỗi năm ảnh hưởng đến sự tăng trưởng của đàn gia súc. Với mỗi năm i , Mansur nhớ một hệ số tăng trưởng là số nguyên dương $X[i]$. Nếu bạn bắt đầu năm i với h con ngựa, cuối năm bạn sẽ có $h \cdot X[i]$ con ngựa trong đàn gia súc của mình.

Các con ngựa chỉ có thể được bán vào cuối năm. Với mỗi năm i , Mansur nhớ một số nguyên dương $Y[i]$: giá một con ngựa anh có thể bán vào cuối năm i . Sau mỗi năm, anh có thể bán một số lượng lớn bất kỳ các con ngựa, mỗi con đều với giá giống nhau $Y[i]$.

Mansur muốn biết số tiền lớn nhất mà hiện nay anh có thể có là bao nhiêu nếu biết cách lựa chọn thời điểm tốt nhất để bán các con ngựa của mình trong N năm qua. Bạn vinh dự được là khách mời trong kỳ *ng nghỉ* (tiếng Kazakhstan là *toi*) của Mansur, và anh ta đề nghị bạn trả lời câu hỏi này.

Trí nhớ của Mansur liên tục cải thiện trong suốt buổi tối, nên anh ta đưa ra 1 dãy gồm M lần cập nhật. Mỗi lần cập nhật sẽ thay đổi một trong các giá trị $X[i]$ hoặc một trong các giá trị $Y[i]$. Sau mỗi lần cập nhật, anh ta lại hỏi bạn số lượng tiền lớn nhất anh ta có thể có nhờ việc bán ngựa của mình. Các lần cập nhật của Mansur có tính tích lũy: mỗi câu trả lời của bạn phải tính đến tất cả các lần cập nhật trước đây. Lưu ý rằng một giá trị $X[i]$ hoặc $Y[i]$ có thể được cập nhật nhiều lần.

Kết quả trả lời cho Mansur có thể là số rất lớn. Để tránh phải làm việc với số lớn, bạn sẽ chỉ phải đưa ra phần dư của phép chia kết quả tìm được cho $10^9 + 7$.

Ví dụ

Giả sử có $N = 3$ năm, với các thông tin sau:

	0	1	2
X	2	1	3
Y	3	4	1

Với các giá trị ban đầu này, Mansur có thể kiếm được nhiều tiền nhất nếu anh ta bán cả hai con ngựa của mình vào cuối năm 1. Toàn bộ quá trình sẽ diễn ra như sau:

- Ban đầu, Mansur có 1 con ngựa.
- Sau năm 0 anh sẽ có $1 \cdot X[0] = 2$ con ngựa.

- Sau năm 1 anh sẽ có $2 \cdot X[1] = 2$ con ngựa.
- Bây giờ anh có thể bán hai con ngựa này. Tổng lợi nhuận thu được sẽ là $2 \cdot Y[1] = 8$.

Sau đó, giả sử có $M = 1$ lần cập nhật: đổi $Y[1]$ thành 2.

Sau lần cập nhật này, chúng ta sẽ có:

	0	1	2
X	2	1	3
Y	3	2	1

Trong trường hợp này, một lời giải tối ưu là bán một con ngựa sau năm 0 và sau đó là ba con ngựa sau năm 2.

Toàn bộ quá trình sẽ diễn ra như sau:

- Ban đầu Mansur có 1 con ngựa.
- Sau năm 0 anh sẽ có $1 \cdot X[0] = 2$ con ngựa.
- Bây giờ anh có thể bán một trong số các con ngựa và thu được $Y[0] = 3$, và còn lại một con ngựa.
- Sau năm 1 anh sẽ có $1 \cdot X[1] = 1$ con ngựa.
- Sau năm 2 anh sẽ có $1 \cdot X[2] = 3$ con ngựa.
- Bây giờ anh có thể bán ba con ngựa này và thu được $3 \cdot Y[2] = 3$. Tổng số tiền là $3 + 3 = 6$.

Nhiệm vụ

Bạn được cho trước N , X , Y , và danh sách các lần cập nhật. Trước lần cập nhật đầu tiên, và sau mỗi lần cập nhật, hãy tính phần dư của phép chia số lượng tiền nhiều nhất mà Mansur có thể kiếm được từ các con ngựa của mình cho $10^9 + 7$. Bạn cần cài đặt các hàm `init`, `updateX`, và `updateY`.

- `init(N, X, Y)` — Chương trình chấm sẽ gọi hàm này đầu tiên và chỉ một lần duy nhất.
 - N : số năm.
 - X : mảng có N phần tử. Với $0 \leq i \leq N - 1$, $X[i]$ là hệ số tăng trưởng của năm i .
 - Y : mảng có N phần tử. Với $0 \leq i \leq N - 1$, $Y[i]$ là giá của một con ngựa sau năm i .
 - Lưu ý rằng cả X và Y chứa các giá trị ban đầu Mansur đưa ra (trước bất kỳ lần cập nhật nào).
 - Sau khi `init` kết thúc, các mảng X và Y vẫn hợp lệ, và bạn có thể thay đổi giá trị của chúng nếu muốn.
 - Hàm này phải trả về phần dư của phép chia số lượng tiền lớn nhất Mansur có thể kiếm được với các giá trị ban đầu của X và Y cho $10^9 + 7$.
- `updateX(pos, val)`

- `pos`: một số nguyên nằm trong khoảng $0, \dots, N - 1$.
 - `val`: giá trị mới của $X[pos]$.
 - Hàm này phải trả về phần dư của phép chia số lượng tiền lớn nhất Mansur có thể kiếm được sau lần cập nhật này cho $10^9 + 7$.
- `updateY(pos, val)`
- `pos`: một số nguyên nằm trong khoảng $0, \dots, N - 1$.
 - `val`: giá trị mới của $Y[pos]$.
 - Hàm này phải trả về phần dư của phép chia số lượng tiền lớn nhất Mansur có thể kiếm được sau lần cập nhật này cho $10^9 + 7$.

Bạn có thể giả sử tất cả các giá trị ban đầu, kể cả giá trị của các lần cập nhật của $X[i]$ và $Y[i]$ đều nằm trong khoảng 1 và 10^9 kể cả hai đầu mút.

Sau khi gọi `init`, chương trình chấm sẽ gọi `updateX` và `updateY` một số lần. Tổng số lần gọi `updateX` và `updateY` là M .

Subtasks

subtask	điểm	N	M	Ràng buộc bổ sung
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	$1 \leq N \leq 1,000$	$0 \leq M \leq 1,000$	không
3	20	$1 \leq N \leq 500,000$	$0 \leq M \leq 100,000$	$X[i] \geq 2$ và $val \geq 2$ tương ứng cho <code>init</code> và <code>updateX</code>
4	23	$1 \leq N \leq 500,000$	$0 \leq M \leq 10,000$	không
5	23	$1 \leq N \leq 500,000$	$0 \leq M \leq 100,000$	không

Chương trình chấm mẫu

Chương trình chấm mẫu đọc dữ liệu đầu vào từ file `horses.in` với định dạng sau:

- dòng 1: N
- dòng 2: $X[0] \dots X[N - 1]$
- dòng 3: $Y[0] \dots Y[N - 1]$
- dòng 4: M
- các dòng 5, ..., $M + 4$: ba số `type pos val` (`type=1` cho `updateX` và `type=2` cho `updateY`).

Chương trình chấm mẫu in ra giá trị trả về của `init`, tiếp theo sau là các giá trị trả về của các lệnh gọi `updateX` và `updateY`.