



## Коники

Манзур любить розводити коней, як і його предки. В нього зараз найбільший табун в Казахстані. Але так було не завжди.  $N$  років тому Манзур був простим джигітом (казахська назва молодого хлопця) і в нього був лише один кінь. Він мріяв заробити багато грошей і врешті стати баєм (казахська назва олігарха).

Пронумеруємо роки від 0 до  $N - 1$  в хронологічному порядку ( $N - 1$  це останній рік). Погода кожного року впливає на ріст табуна. Для кожного року  $i$ , Манзур пам'ятає додатне ціле число — коефіцієнт росту табуна  $X[i]$ . Якщо ви почали рік  $i$  з  $h$  кіньми, то закінчите його з  $h \cdot X[i]$  кіньми в табуні.

Коней можна продавати лише в кінці року. Для кожного року  $i$ , Манзур пам'ятає додатне ціле число  $Y[i]$ : ціну по якій можна було продати коня в кінці року  $i$ . В кінці кожного року можна було продати як завгодно багато коней по ціні  $Y[i]$ .

Манзуру цікаво яку найбільшу кількість грошей він міг би заробити, якщо б обирав оптимальні моменти для продажу коней протягом останніх  $N$  років. Вам випала честь бути запрошеними на свято Манзура і він попросив вас дати відповідь на це питання.

Протягом вечора пам'ять Манзура почала покрашуватись і він почав давати вам послідовно  $M$  уточнень. Кожне уточнення змінює лише одну з величин  $X[i]$  або  $Y[i]$ . Після кожного уточнення він знову питає вас найбільшу кількість грошей, що міг би заробити. Уточнення Манзура кумулятивні, тобто відповідь на кожне наступне питання має враховувати всі попередні уточнення. Зауважимо, що кожна із величин  $X[i]$  та  $Y[i]$  може уточнюватись кілька разів.

Точні відповіді на запитання Манзура можуть бути величезними. Для того щоб уникнути роботи з великими числами досить просто давати відповідь по модулю  $10^9 + 7$ .

## Приклад

Припустимо  $N = 3$  роки з наступною інформацією:

|   |   |   |   |
|---|---|---|---|
|   | 0 | 1 | 2 |
| X | 2 | 1 | 3 |
| Y | 3 | 4 | 1 |

Для цих початкових даних Манзур може заробити найбільше грошей, якщо продасть обох коней в кінці року 1. Весь процес буде виглядати наступним чином:

- Спочатку в Манзура один кінь.
- Після року 0 в нього  $1 \cdot X[0] = 2$  коня.

- Після року 1 в нього  $2 \cdot X[1] = 2$  коня.
- Тепер він може продати обидва коня. Загальний прибуток буде  $2 \cdot Y[1] = 8$ .

Тепер припустимо, що є одне уточнення: заміна  $Y[1]$  на 2.

Після доповнення маємо:

|   |   |   |   |
|---|---|---|---|
|   | 0 | 1 | 2 |
| X | 2 | 1 | 3 |
| Y | 3 | 2 | 1 |

В цьому випадку оптимальна стратегія – продати одного коня після року 0, а потім трьох після року 2.

Весь процес буде виглядати наступним чином:

- Спочатку в Манзура 1 кінь.
- Після року 0 в нього буде  $1 \cdot X[0] = 2$  коня.
- Він може продати одного з цих коней за  $Y[0] = 3$ , і залишити ще одного.
- Після року 1 в нього буде  $1 \cdot X[1] = 1$  кінь.
- Після року 2 в нього буде  $1 \cdot X[2] = 3$  коня.
- Тепер цих 3 коней можна продати за  $3 \cdot Y[2] = 3$ . Загальна кількість зароблених грошей  $3+3=6$ .

## Задача

Дано  $N$ ,  $X$ ,  $Y$  і список уточнень. До першого уточнення і після кожного уточнення обчисліть максимальну кількість грошей, що міг би заробити Манзур по модулю  $10^9 + 7$ .

Вам потрібно написати функції `init`, `updateX` і `updateY`.

- `init(N, X, Y)` — Модуль перевірки викличе цю функцію першою і лише один раз.
  - $N$ : кількість років.
  - $X$ : масив довжини  $N$ . Для  $0 \leq i \leq N - 1$ ,  $X[i]$  — коефіцієнт росту табуна для року  $i$ .
  - $Y$ : масив довжини  $N$ . Для  $0 \leq i \leq N - 1$ ,  $Y[i]$  — ціна продажу коня після року  $i$ .
  - Зверніть увагу, що величини  $X$  і  $Y$  показують початкові дані, що повідомляє нам Манзур (до будь-яких уточнень).
  - Після того, як `init` завершить свою роботу, масиви  $X$  та  $Y$  залишаються доступними і ви можете вносити в них потрібні вам зміни.
  - Функція має повернути максимальну кількість грошей, що може заробити Манзур для початкових даних  $X$  і  $Y$ , по модулю  $10^9 + 7$ .
- `updateX(pos, val)`

- `pos`: ціле число з діапазону  $0, \dots, N - 1$ .
  - `val`: нове значення величини  $X[pos]$ .
  - Функція повинна повернути максимальну кількість грошей, що може заробити Манзур після цього уточнення по модулю  $10^9 + 7$ .
- `updateY(pos, val)`
- `pos`: ціле число з діапазону  $0, \dots, N - 1$ .
  - `val`: нове значення величини  $Y[pos]$ .
  - Функція повинна повернути максимальну кількість грошей, що може заробити Манзур після цього уточнення по модулю  $10^9 + 7$ .

Можемо припустити, що всі початкові дані, як і всі доповнення  $X[i]$  і  $Y[i]$  лежать в межах від 1 до  $10^9$  включно.

Після виклику `init`, модуль перевірки викличе `updateX` і `updateY` кілька разів. Загальна кількість викликів функцій `updateX` і `updateY` буде  $M$ .

## Підзадачі

| Підзадача | Бали | $N$                     | $M$                     | Додаткові умови   |
|-----------|------|-------------------------|-------------------------|---|
| 1         | 17   | $1 \leq N \leq 10$      | $M = 0$                 | $X[i], Y[i] \leq 10$ ,<br>$X[0] \cdot X[1] \cdot \dots \cdot X[N - 1] \leq 1,000$     |
| 2         | 17   | $1 \leq N \leq 1,000$   | $0 \leq M \leq 1,000$   | відсутні  |
| 3         | 20   | $1 \leq N \leq 500,000$ | $0 \leq M \leq 100,000$ | $X[i] \geq 2$ і $val \geq 2$ для <code>init</code> та <code>updateX</code> відповідно |
| 4         | 23   | $1 \leq N \leq 500,000$ | $0 \leq M \leq 10,000$  | відсутні  |
| 5         | 23   | $1 \leq N \leq 500,000$ | $0 \leq M \leq 100,000$ | відсутні  |

## Приклад модуля перевірки

Отриманий вами модуль перевірки зчитує вхідні дані з файла `horses.in` в наступному форматі:

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

Модуль перевірки виводить значення функції `init` а потім значення всіх функцій `updateX` і `updateY`.