

Lovak

Mansur az utóbbi N évben tenyésztett lovakat. Kezdetben egy lova volt. Az i -edik évben $X[i]$ -szeresére nőtt az év végére a lovai száma, ahol $X[i]$ pozitív egész szám. Tehát ha az i -edik év elején lovai száma h , akkor az év végére $h \cdot X[i]$ lova lett.

Lovak eladásával sok pénzt akar keresni. Az i -edik évben egy ló ára $Y[i]$. Minden évben akárhány lovat eladhat.

Mansur a lehető legtöbbet akarja keresni a lovak eladásából, amihez meg kell határozni, hogy mikor és hány lovat adjon el!

Mansur arra is kíváncsi, hogy ha az $X[i]$ vagy az $Y[i]$ megváltozik, akkor mennyi lenne a legtöbb bevétel? Ez a változás M alkalommal következhet be, a legtöbb bevételt kezdetben és minden változás után is meg kell adni! A változások kumulatívak, tehát a korábbi változásokat is figyelembe kell venni! Az eredmény nagyon nagy is lehet, ezért moduló $10^9 + 7$ kell megadni!

Példa

Legyen $N = 3$ év a következő információkkal:

	0	1	2
X	2	1	3
Y	3	4	1

Ezekre a kezdőértékekre akkor lesz a legtöbb bevétel, ha két lovat ad el az 1. év végén.

- kezdetben 1 lova van;
- a 0. év végén $1 \cdot X[0] = 2$ lova lesz;
- az 1. év végén $2 \cdot X[1] = 2$ lova lesz;
- ha most eladja a két lovat, akkor a bevétel $2 \cdot Y[1] = 8$ lesz.

Legyen ezután $M = 1$ és az $Y[1]$ változzon 2-re!

A módosítás után:

	0	1	2
X	2	1	3
Y	3	2	1

Ekkor a legtöbb bevétel úgy is keletkezhet, hogy egy lovat a 0. év végén, három lovat pedig a 2. év végén ad el.

- kezdetben 1 lova van

- a 0. év végén $1 \cdot X[0] = 2$ lova lesz.
- egyet elad $Y[0] = 3$ összegért, marad 1 lova.
- az 1. év végén $1 \cdot X[1] = 1$ lova lesz
- a 2. év végén $1 \cdot X[2] = 3$ lova lesz
- ebben az évben a három ló eladásával $3 \cdot Y[2] = 3$ összeget keres. A teljes bevétel $3 + 3 = 6$

Feladat

Adottak N , X , Y és a módosítások. A módosítások előtt és minden módosítás után számold ki, hogy mennyi lehet Mansur maximális bevétele, moduló $10^9 + 7$!

Az `init`, `updateX` és `updateY` függvényeket kell megírnod!

- `init(N, X, Y)` — Az értékelő ezt hívja először, egyszer.
 - N : az évek száma.
 - X : N elemű tömb, minden $0 \leq i \leq N - 1$ -re $X[i]$ az i -edik év növekedés szorzótényezője.
 - Y : N elemű tömb, minden $0 \leq i \leq N - 1$ -re $Y[i]$ egy ló ára az i -edik év végén.
 - Megjegyezzük, hogy X és Y a változások előtti értékeket tartalmazza.
 - A függvény visszaadott értéke a maximális bevétel legyen moduló $10^9 + 7$, amit Mansur kaphat!
- `updateX(pos, val)`
 - `pos`: $0, \dots, N - 1$ közötti egész szám.
 - `val`: az $X[pos]$ új értéke.
 - A függvény visszaadott értéke a maximális bevétel legyen moduló $10^9 + 7$ a módosítás után!
- `updateY(pos, val)`
 - `pos`: $0, \dots, N - 1$ közötti egész szám.
 - `val`: az $Y[pos]$ új értéke.
 - A függvény visszaadott értéke a maximális bevétel legyen moduló $10^9 + 7$ a módosítás után!

Feltehető, hogy $X[i]$ és $Y[i]$ kezdőértékei és módosítás utáni értékei is 1 és 10^9 közöttiek.

Az `init` hívása után az értékelő összesen M módosítást hív, mindegyik vagy az `updateX`, vagy az `updateY`.

Részfeladatok

részfeladat	pontok	N	M	további korlátok
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	$1 \leq N \leq 1,000$	$0 \leq M \leq 1,000$	nincs
3	20	$1 \leq N \leq 500,000$	$0 \leq M \leq 100,000$	$X[i] \geq 2$ és $val \geq 2$ az init-ben és az updateX-ben
4	23	$1 \leq N \leq 500,000$	$0 \leq M \leq 10,000$	nincs
5	23	$1 \leq N \leq 500,000$	$0 \leq M \leq 100,000$	nincs

Megvalósítás

Egyetlen `horses.c`, `horses.cpp`, `horses.pas`, vagy `horses.java` file-t kell beadnod, ami a megoldás függvényeit tartalmazza!

C/C++ program (include `horses.h` legyen a file elején)

```
int init(int N, int X[], int Y[]);
int updateX(int pos, int val);
int updateY(int pos, int val);
```

Pascal programs (implement the described method in the unit `horses`)

```
function init(N : longint; var X, Y : array of longint) : longint;
function updateX(pos, val : longint) : longint;
function updateY(pos, val : longint) : longint;
```

Minta értékelő

A `horses.in` file-ból olvas a következő formában

- Az 1. sor: N
- A 2. sor: $X[0] \dots X[N-1]$
- A 3. sor: $Y[0] \dots Y[N-1]$
- A 4. sor: M
- Az 5., ..., $M+4$. sorokban 3 szám van: `type pos val` (`type=1` `updateX` esetén és `type=2` `updateY` esetén).

A minta értékelő kiírja az `init`, valamint az `updateX` és `updateY` függvények értékeit.