



## Šváby

Šandyna rada chová šváby. Prečo to robí? Netuším. Keď sa jej na to spýtate, buď vám povie, že sú to prekrásne a inteligentné chrobáčky (a navyše veľmi oddolné, takže až tak nevádi, keď ich zabudnete nakŕmiť), alebo že sa s nimi robí dobrý biznis. Stále som si nie istý, kto dobrovoľne kúpi švába, ale keďže Šandyna je už ich najväčšou chovateľkou, nezostáva mi iné, iba jej veriť. (Mišof hovorí, že šváby kupujú chovatelia škorpiónov.)

Celý Šandynin chovateľský podnik začal pred  $N$  rokmi, keď získala svojho prvého švába. V rôzne roky bola rôzna klíma, a to malo vplyv na to, ako rýchlo s Šandynine šváby množili. Očíslujme posledných  $N$  rokov číslami od  $0$  po  $N - 1$ , pričom nultý rok bol najdávnejšie. Ak mala Šandyna na začiatku  $i$ -teho roka  $h$  švábov, na jeho konci ich mala  $h \cdot X[i]$ , kde  $X[i]$  je kladné celé číslo označujúce koeficient rastu v roku  $i$ . Všimnite si, že na začiatku roku  $0$  mala jedného švába, teda bolo  $h = 1$ .

Na konci každého roka sa navyše konala veľká švábia burza, na ktorej mohla Šandyna predat niekoľko zo svojich švábov. Predané množstvo švábov mohlo byť ľubovoľné, nemusela predat žiadneho, ale mohla predat aj všetkých. Jediné, čo sa naprieč rokmi menilo, bola cena za jedného švába. Na konci  $i$ -teho roka sa jeden šváb predával za  $Y[i]$  peňazí, pričom hodnoty  $Y[i]$  boli kladné celé čísla.

Šandynu by teraz zaujímalo, koľko peňazí mohla počas posledných  $N$  rokov zarobiť, ak by predávala švábov v najlepších možných časoch.

Z intershipu v Googli sa však práve vrátil Jaro a vysvitlo, že si to Šandyna pamätá celé zle. Veľa hodnôt  $X[i]$  a  $Y[i]$  je zlých, preto ich Jaro postupne opravuje. Po každej zmene nejakej z hodnôt chce Šandyna opäť vedieť, aký mohol byť jej maximálny zisk. Jaro spraví postupne  $M$  úprav. Pri každej úprave zmení buď jednu z hodnôt  $X[i]$  alebo jednu z hodnôt  $Y[i]$ . Jarove zmeny sú inkrementálne: každá z nich nadväzuje na predchádzajúcu, ako keby Jaro postupne prepisoval hodnoty v poliach  $X$  a  $Y$ . Jedna hodnota  $X[i]$  alebo  $Y[i]$  môže byť Jarom postupne zmenená aj viackrát.

Samozrejme, množstvo peňazí, ktoré mohla Šandyna zarobiť, môže byť obrovské (je to naozaj výhodný biznis), preto má váš program určiť len zvyšok, ktorý táto hodnota dáva po delení číslom  $10^9 + 7$ .

## Príklad

Predpokladajme, že  $N = 3$  a hodnoty v  $X$  a  $Y$  vyzerajú nasledovne:

	0	1	2
X	2	1	3
Y	3	4	1

Pre tieto hodnoty zarobí Šandyna najviac peňazí vtedy, keď predá oba svoje šváby na konci roka 1.

Bude to prebiehať takto:

- Na začiatku roka **0** má Šandyna **1** švába.
- Na konci roka **0** bude mať  $1 \cdot X[0] = 2$  švábov.
- Na konci roka **1** bude mať  $2 \cdot X[1] = 2$  švábov.
- Oba šváby predá, čím získa  $2 \cdot Y[1] = 8$  peňazí.
- V roku **2** už nemá žiadne šváby, teda sa jej žiadne nerozmnožia.
- Dohromady zarobí **8** peňazí.

Potom príde Jaro a urobí postupne  $M$  zmien. Predpokladajme, že  $M = 1$  a v jedinej zmene Jaro zmení hodnotu  $Y[1]$  na **2**. To znamená, že po zmene polia  $X$  a  $Y$  vyzerajú takto:

	<b>0</b>	<b>1</b>	<b>2</b>
X	2	1	3
Y	3	2	1

Po tejto úprave je jednou z optimálnych možností predat' jedného švába na konci roka **0** a potom všetkých **3** švábov na konci roka **2**. Bude to prebiehať takto:

- Na začiatku roka **0** má Šandyna **1** švába.
- Na konci roka **0** bude mať  $1 \cdot X[0] = 2$  švábov.
- Na konci roka **0** jedného z nich predá za cenu  $Y[0] = 3$ .
- Na začiatku roka **1** má teda **1** švába.
- Na konci roka **1** bude mať  $1 \cdot X[1] = 1$  švába.
- Na konci roka **2** bude mať  $1 \cdot X[2] = 3$  šváby.
- Na konci roka **2** predá všetky tri šváby za cenu  $3 \cdot Y[2] = 3$ .
- Dohromady zarobí  $3 + 3 = 6$  peňazí.

## Úloha

Dostanete hodnoty  $N$ ,  $X$ ,  $Y$  a zoznam Jarových zmien. Aj úplne na začiatku (pred všetkými zmenami) a potom aj po každej zmene vypočítajte maximálne množstvo peňazí, ktoré vie Šandyna zarobiť predávaním švábov, modulo  $10^9 + 7$ .

Na riešenie úlohy musíte implementovať funkcie `init()`, `updateX()` and `updateY()`.

- `init(N, X, Y)` — grader zavolá túto funkciu práve raz, a to na začiatku programu.
  - $N$ : počet rokov.
  - $X$ : pole dĺžky  $N$ . Hodnoty  $X[i]$  (pre  $0 \leq i \leq N - 1$ ) udávajú pre každý rok koeficient rastu počtu švábov.
  - $Y$ : pole dĺžky  $N$ . Hodnoty  $Y[i]$  (pre  $0 \leq i \leq N - 1$ ) udávajú pre každý rok cenu za

jedného švába.

- Polia  $X$  a  $Y$  predstavujú počiatočné hodnoty udané Šandynou, pred ľubovoľnou Jarovou zmenou.
- Potom ako sa funkcia `init()` ukončí, polia  $X$  a  $Y$  zostanú v pamäti a môžete meniť ich obsah.
- Funkcia musí vrátiť maximálne množstvo peňazí, ktoré vie Šandyna zarobiť, modulo  $10^9 + 7$ .
- `updateX(pos, val)`
  - `pos`: celé číslo z intervalu  $0$  až  $N - 1$ .
  - `val`: nová hodnota  $X[pos]$ .
  - Funkcia musí vrátiť maximálne množstvo peňazí, ktoré vie Šandyna zarobiť po tejto zmene, modulo  $10^9 + 7$ .
- `updateY(pos, val)`
  - `pos`: celé číslo z intervalu  $0$  až  $N - 1$ .
  - `val`: nová hodnota  $Y[pos]$ .
  - Funkcia musí vrátiť maximálne množstvo peňazí, ktoré vie Šandyna zarobiť po tejto zmene, modulo  $10^9 + 7$ .

Môžete predpokladať, že pôvodné aj upravené hodnoty  $X[i]$  a  $Y[i]$  sú celé čísla od  $1$  po  $10^9$  vrátane. Po zavolaní funkcie `init()` zavolá grader niekoľkokrát funkcie `updateX()` a `updateY()`. Dokopy zavolá funkcie `updateX()` a `updateY()` práve  $M$ -krát.

## Podúlohy

podúloha	body	$N$	$M$	dodatočné obmedzenia
1	17	$1 \leq N \leq 10$	$M = 0$	$X[i], Y[i] \leq 10$ , $X[0] \cdot X[1] \cdot \dots \cdot X[N - 1] \leq 1000$
2	17	$1 \leq N \leq 1000$	$0 \leq M \leq 1000$	žiadne
3	20	$1 \leq N \leq 500\,000$	$0 \leq M \leq 100\,000$	$X[i] \geq 2$ a $val \geq 2$ pre <code>init()</code> a <code>updateX()</code>
4	23	$1 \leq N \leq 500\,000$	$0 \leq M \leq 10\,000$	žiadne
5	23	$1 \leq N \leq 500\,000$	$0 \leq M \leq 100\,000$	žiadne

## Sample grader

Sample grader číta vstup zo súboru `horses.in`, ktorý je v nasledovnom formáte:

- riadok 1:  $N$
- riadok 2:  $X[0] \dots X[N - 1]$

- riadok 3:  $Y[0] \dots Y[N - 1]$
- riadok 4:  $M$
- riadky 5, ...,  $M + 4$ : tri čísla — `type pos val` (`type=1` pre `updateX()` a `type=2` pre `updateY()`).

Sample grader vypíše návratovú hodnotu funkcie `init()` a potom návratové hodnoty funkcií `updateX()` a `updateY()` v poradí v akom sa objavili na vstupe.