

Beygirler

Mansur, ataları gibi, beygir yetiştirmeyi seven biridir. Şu anda Kazakistan'ın en büyük sürüsü ona aittir. Fakat önceki yıllarda zengin değildi. N yıl önce, Mansur genç birisi iken sadece 1 beygire sahipti ve gelecekte çok para kazanarak ülkenin önemli ağalarından biri olmayı düşlüyordu.

Yılları 0 'dan $N - 1$ 'e kadar kronolojik numaralandırılım (Yani, yıl $N - 1$ şimdiki zamana en yakındır). Her yılın hava durumu sürünün büyümesini etkilemektedir. Her bir i yılı için, Mansur sürünün büyüme katsayısı $X[i]$ 'yi (bir pozitif tamsayı) hatırlamaktadır. Eğer i yılına h tane beygirle başlanırsa yıl sonunda sürüde $h \cdot X[i]$ tane beygir olmaktadır.

Beygirler sadece yıl sonlarında satılabilmektedir. Her bir i yılı için, Mansur bir beygirin satış fiyatı $Y[i]$ 'yi de (bir pozitif tamsayı) hatırlamaktadır. Her bir yılın sonunda sürüden istenen sayıda beygir satılabilmektedir. i yılı sonu için her bir beygirin satış fiyatı aynı olup $Y[i]$ liradır.

Mansur sürüdeki beygirlerini en uygun zamanda satarak N yılın sonunda maksimum parayı kazanmak istemektedir. Kendisine yardımcı olmanız için Mansur sizi kendi yaylasına davet etmiştir.

Mansur'un hafızası akşamleyin kendine geldiğinde M tane güncelleme yapar. Her bir güncelleme ya $X[i]$ ya da $Y[i]$ değerlerinden yalnız birini değiştirmektedir. Her bir güncelleme sonrası size tekrar aynı sorunun (beygirlerini en uygun zamanda satarak elde edebileceği maksimum kazanç) cevabını sormaktadır. Yeni güncellemeler öncekilerin üzerine yığılmalı yapılmaktadır. Yani, her bir yeni güncelleme kendinden önceki tüm güncellemelerin devamı niteliğindedir. Herhangi bir $X[i]$ ya da $Y[i]$ değeri çok kez güncellenebilir.

Mansur'un sorusunun cevabı çok büyük sayılar olabileceğinden cevabın $10^9 + 7$ 'ye göre modunu döndürmeniz gerekir.

Örnek

$N = 3$ yıl ve yıllara göre ilk X, Y değerleri aşağıdaki gibi verilmiş olsun:

	0	1	2
X	2	1	3
Y	3	4	1

İlk değerler için, Mansur yıl 1 sonunda tüm beygirlerini satarsa maksimum parayı kazanır. Detaylar şu şekildedir:

- Başlangıçta Mansur'un 1 beygiri vardır.
- Yıl 0 sonunda $1 \cdot X[0] = 2$ beygiri olur.
- Yıl 1 sonunda $2 \cdot X[1] = 2$ beygiri olur.

- Bu beygirleri yıl 1 sonunda satarak $2 \cdot Y[1] = 8$ lira kazanır.

Güncelleme sayısı $M = 1$ olarak verilsin: bu güncelleme $Y[1]$ değerini 2 yapmış olsun. Güncelleme sonrası tablo şu şekildedir.

	0	1	2
X	2	1	3
Y	3	2	1

Bu durumda optimal çözümlerden birisi şöyledir: beygirlerden birini yıl 0 sonunda diğer üçünü ise yıl 2 sonunda satar. Detaylar şöyledir:

- Başlangıçta Mansur'un 1 beygiri vardır.
- Yıl 0 sonunda $1 \cdot X[0] = 2$ beygiri olur.
- Beygirlerden birini $Y[0] = 3$ liraya satar ve elinde 1 beygir kalır.
- Yıl 1 sonunda $1 \cdot X[1] = 1$ beygiri olur.
- Yıl 2 sonunda $1 \cdot X[2] = 3$ beygiri olur.
- Bu üç beygiri $3 \cdot Y[2] = 3$ liraya satar. Toplamda $3 + 3 = 6$ lira kazanır.

Görev

Size N , X , Y ve güncelleme listesi verilecektir. İlk güncelleme öncesi ve her bir güncelleme sonrası Mansur'un kazanabileceği maksimum para miktarı mod $10^9 + 7$ bulmalısınız. Üç fonksiyon `init`, `updateX` ve `updateY` gerçekleştirmeniz gerekir.

- `init(N, X, Y)` — Grader ilk olarak bu fonksiyonu bir kez çağıracaktır.
 - N : yıl sayısı.
 - X : N uzunluğunda bir dizi. Her bir $0 \leq i \leq N - 1$ yılı için $X[i]$ o yılda sürünün büyüme katsayısıdır.
 - Y : N uzunluğunda bir dizi. Her bir $0 \leq i \leq N - 1$ yılı için $Y[i]$ o yılın sonu için bir beygirin satış fiyatıdır.
 - X ve Y değerleri herhangi bir güncelleme öncesi Mansur tarafından belirlenen ilk değerlerdir.
 - `init` bittikten sonra, X ve Y dizileri geçerli olmaya devam eder, bunların içeriğini daha sonra isterseniz değiştirebilirsiniz.
 - Fonksiyon, Mansur'un X ve Y ilk değerleri ile kazanabileceği maksimum para miktarının $10^9 + 7$ 'ye göre modunu döndürmelidir.
- `updateX(pos, val)`
 - `pos`: $0, \dots, N - 1$ aralığında bir tamsayı.
 - `val`: $X[pos]$ 'un yeni değeri.

- Fonksiyon, Mansur'un bu güncelleme sonrası kazanabileceği maksimum para miktarının $10^9 + 7$ 'ye göre modunu döndürmelidir.
- `updateY(pos, val)`
 - `pos`: $0, \dots, N - 1$ aralığında bir tamsayı.
 - `val`: $Y[pos]$ 'un yeni değeri.
 - Fonksiyon, Mansur'un bu güncelleme sonrası kazanabileceği maksimum para miktarının $10^9 + 7$ 'ye göre modunu döndürmelidir.

Her bir $X[i]$ ve $Y[i]$ için ilk ve güncelleme sonrası değerlerin 1 ile 10^9 arasında olacağını kabul edebilirsiniz.

`init`'i çağırdıktan sonra, grader `updateX` ve `updateY` fonksiyonlarını çok kez çağırabilir. `updateX` ve `updateY`'nin toplam çağırılma sayısı M olacaktır.

Alt görevler

alt görev	puan	N	M	ilave kısıtlar
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	$1 \leq N \leq 1,000$	$0 \leq M \leq 1,000$	yok
3	20	$1 \leq N \leq 500,000$	$0 \leq M \leq 100,000$	<code>init</code> için $X[i] \geq 2$ ve <code>updateX</code> için $val \geq 2$
4	23	$1 \leq N \leq 500,000$	$0 \leq M \leq 10,000$	yok
5	23	$1 \leq N \leq 500,000$	$0 \leq M \leq 100,000$	yok

Örnek grader

Örnek grader girdiyi `horses.in` dosyasından aşağıdaki formatta okur:

- satır 1: N
- satır 2: $X[0] \dots X[N - 1]$
- satır 3: $Y[0] \dots Y[N - 1]$
- satır 4: M
- satırlar 5, ..., $M + 4$: üç sayı `type pos val` (`type=1` değeri `updateX` ve `type=2` değeri `updateY` belirler).

Örnek grader `init` fonksiyonundan ve takip eden `updateX` ve `updateY` fonksiyonlarından dönen değerleri yazdırır.