



## ฟาร์มม้า

มันเซอร์รักการทำฟาร์มม้าเช่นเดียวกับบรรพบุรุษของเขา ปัจจุบันเขามีม้ามากที่สุดในคาซัคสถาน แต่เดิมที่เขาไม่ได้มีม้าเลยขนาดนั้น เมื่อ  $N$  ปีที่แล้ว เขาเป็นเพียง dzhigit (ภาษาคาซัคแปลว่า *เด็กหนุ่ม*) และเขาก็มีม้าเพียงตัวเดียว ณ ตอนนั้นเขาว่าฝันที่จะเป็น bai (ภาษาคาซัคแปลว่า *มหาเศรษฐี*)

เราจะกำกับตัวเลขปีโดยเริ่มจาก 0 ไปจนถึง  $N-1$  ตามลำดับก่อนหลัง (ปี  $N-1$  คือปีล่าสุด) สภาพอากาศในแต่ละปีมีผลต่อการเจริญเติบโตของม้าในคอกของเขา สำหรับปีที่  $i$  มันเซอร์จำค่าสัมประสิทธิ์การเติบโต (growth coefficient) ได้ ซึ่งแทนด้วยจำนวนเต็มบวก  $X[i]$  ดังนั้นหากเขาเริ่มปีที่  $i$  ด้วยม้าจำนวน  $h$  ตัว เขาก็จะมีม้าเป็นจำนวน  $h \cdot X[i]$  ตอนสิ้นปีนั้น

การขายม้าจะเกิดขึ้นได้ตอนสิ้นปีเท่านั้น สำหรับปีที่  $i$  มันเซอร์จำราคาที่เขาสามารถขายม้าหนึ่งตัวตอนสิ้นปีที่  $i$  ได้ ซึ่งแทนด้วยจำนวนเต็มบวก  $Y[i]$  โดยสำหรับปีที่  $i$  เขาจะขายม้ากี่ตัวก็ได้ ซึ่งจะขายแต่ละตัวในราคาเดียวกัน คือ  $Y[i]$

มันเซอร์ต้องการทราบว่าเขาจะสามารถได้เงินได้มากที่สุดเท่าไรหากเขาเลือกขายม้าในคอก ณ จังหวะที่ดีที่สุดตลอด  $N$  ปีที่ผ่านมา เนื่องจากคุณได้รับเกียรติเป็นแขกของเขาระหว่าง toi (ภาษาคาซัคแปลว่า *วันหยุด*) เขาจึงขอให้คุณช่วยหาคำตอบให้

ตลอดช่วงเย็นที่คุยกันนั้น มันเซอร์เริ่มจำรายละเอียดได้มากขึ้นทีละนิด จึงเป็นที่มาของลำดับการแก้ไขทั้งสิ้น  $M$  ครั้ง แต่ละครั้งจะเปลี่ยนค่า  $X[i]$  หรือ  $Y[i]$  เพียงค่าใดค่าหนึ่งเท่านั้น หลังจากการแก้ไขแต่ละครั้งเขาจะถามคุณใหม่ว่า เขาสามารถได้เงินมากที่สุดเท่าไรจากการขายม้าของเขา สังเกตว่าการแก้ไขของมันเซอร์เป็นการแก้ไขแบบสะสม กล่าวคือในการคำนวณคำตอบของคุณจะต้องรวมทุกการแก้ไขก่อนหน้าทั้งหมด นอกจากนี้คุณควรเข้าใจว่าแต่ละ  $X[i]$  หรือ  $Y[i]$  อาจถูกแก้ไขได้หลายครั้ง

คำตอบที่แท้จริงของคำถามที่มันเซอร์ต้องการอาจมีขนาดใหญ่มาก เพื่อหลีกเลี่ยงการประมวลผลเลขขนาดใหญ่ คำตอบของคุณให้ตอบเป็นเศษจากการหารคำตอบจริงด้วย  $10^9 + 7$  (คำตอบ modulo  $10^9 + 7$ )

## ตัวอย่าง

หากมีทั้งหมด  $N = 3$  ปี พร้อมด้วยข้อมูลดังต่อไปนี้

	0	1	2
X	2	1	3
Y	3	4	1

สำหรับค่าเริ่มต้นดังตาราง มันเซอร์จะได้เงินมากที่สุดหากเขาขายม้าทั้งคู่ตอนสิ้นปีที่ 1 กระบวนการทั้งหมดเป็นดังนี้

- เริ่มแรกมันเซอร์มีม้า 1 ตัว
- หลังจากปีที่ 0 เขาจะมีม้า  $1 \cdot X[0] = 2$  ตัว
- หลังจากปีที่ 1 เขาจะมีม้า  $2 \cdot X[1] = 2$  ตัว
- ณ ตอนนั้นเขาสามารถขายม้า 2 ตัว ได้เงินทั้งหมด  $2 \cdot Y[1] = 8$

คราวนี้หากมีการปรับเปลี่ยนทั้งหมด  $M = 1$  ครั้ง โดยเปลี่ยน  $Y[1]$  เป็น 2 ซึ่งแปลว่าหลังจากการเปลี่ยนแปลงนี้ข้อมูลของเขาจะเป็นดังนี้

	0	1	2
X	2	1	3
Y	3	2	1

ในกรณีนี้ หนึ่งในคำตอบที่ดีที่สุดได้มาโดยการขายม้า 1 ตัวตอนสิ้นปีที่ 0 และขายม้า 3 ตัวเมื่อสิ้นปีที่ 2

ดังนั้นกระบวนการทั้งหมดสรุปได้ดังนี้

- เริ่มแรกมันเซอร์มีม้า 1 ตัว
- หลังจากปีที่ 0 เขาจะมีม้า  $1 \cdot X[0] = 2$  ตัว
- ตอนนั้นเขาขายม้า 1 ตัวในราคา  $Y[0] = 3$  และมีม้าเหลืออยู่หนึ่งตัว
- หลังจากปีที่ 1 เขาจะมีม้า  $1 \cdot X[1] = 1$  ตัว
- หลังจากปีที่ 2 เขาจะมีม้า  $1 \cdot X[2] = 3$  ตัว
- ณ ตอนนั้นเขาสามารถขายม้า 3 ตัวเป็นเงิน  $3 \cdot Y[2] = 3$   
ดังนั้นจำนวนเงินที่เขาได้รับทั้งหมดคือ  $3 + 3 = 6$

## งานของคุณ

คุณจะได้รับค่า  $N$ ,  $X$ ,  $Y$  และรายการแก้ไข จากนั้นก่อนหน้าการแก้ไขครั้งแรกและสำหรับทุกๆ ครั้งของการแก้ไขข้อมูล จงคำนวณเงินที่มากที่สุดที่มันเซอร์จะได้จากการขายม้าโดยหาเศษที่เหลือจากการหารด้วย  $10^9 + 7$  ในการนี้คุณต้องเขียนฟังก์ชัน `init`, `updateX` และ `updateY`

- `init(N, X, Y)` — เกรดเดอร์จะเรียกฟังก์ชันนี้เป็นสิ่งแรกและเรียกเพียงครั้งเดียว
  - $N$ : จำนวนปี
  - $X$ : อาร์เรย์ความยาว  $N$  โดยสำหรับ  $0 \leq i \leq N-1$  ค่าของ  $X[i]$  คือสัมประสิทธิ์การเติบโต (growth coefficient) สำหรับปีที่  $i$
  - $Y$ : อาร์เรย์ความยาว  $N$  โดยสำหรับ  $0 \leq i \leq N-1$  ค่าของ  $Y[i]$  คือราคาขายของม้าหนึ่งตัว ณ ปลายปีที่  $i$
  - สังเกตว่าค่า  $X$  และ  $Y$  ที่ให้นี้เป็นค่าเริ่มต้นก่อนการแก้ไขใดๆ
  - หลังจาก `init` สิ้นสุด อาร์เรย์ทั้ง  $X$  และ  $Y$  จะยังคงอยู่ และคุณสามารถแก้ไขค่าของมันได้ถ้าต้องการ
  - ฟังก์ชันนี้ต้องคืนจำนวนเงินที่มากที่สุดที่มันเซอร์จะได้จากการขายม้าโดยหาเศษที่เหลือจากการหารด้วย  $10^9 + 7$
- `updateX(pos, val)`
  - `pos`: เลขจำนวนเต็มจากช่วง  $0, \dots, N-1$ .
  - `val`: ค่าใหม่ของ  $X[pos]$
  - ฟังก์ชันนี้ต้องคืนจำนวนเงินที่มากที่สุดที่มันเซอร์จะได้จากการขายม้าโดยหาเศษที่เหลือจากการหารด้วย  $10^9 + 7$  โดยเป็นค่าหลังการแก้ไขปัจจุบันแล้ว
- `updateY(pos, val)`
  - `pos`: เลขจำนวนเต็มจากช่วง  $0, \dots, N-1$ .
  - `val`: ค่าใหม่ของ  $Y[pos]$
  - ฟังก์ชันนี้ต้องคืนจำนวนเงินที่มากที่สุดที่มันเซอร์จะได้จากการขายม้าโดยหาเศษที่เหลือจากการหารด้วย  $10^9 + 7$  โดยเป็นค่าหลังการแก้ไขปัจจุบันแล้ว

ในโจทย์ข้อนี้ ค่าของ  $X[i]$ ,  $Y[i]$  ทั้งตอนเริ่มต้นและตลอดการแก้ไข จะเป็นค่าในช่วง 1 ถึง  $10^9$

(รวมเลขนี้ด้วย) หลังจากเรียก `init` เกรดเดอร์จะเรียก `updateX` และ `updateY` เป็นจำนวนทั้งหมด  $M$  ครั้ง

## ปัญหาย่อย

ปัญหาย่อย	คะแนน	$N$	$M$	ข้อจำกัดเพิ่มเติม
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	$1 \leq N \leq 1,000$	$0 \leq M \leq 1,000$	ไม่มีเพิ่มเติม
3	20	$1 \leq N \leq 500,000$	$0 \leq M \leq 100,000$	$X[i] \geq 2$ and $val \geq 2$ สำหรับ <code>init</code> และ <code>updateX</code>
4	23	$1 \leq N \leq 500,000$	$0 \leq M \leq 10,000$	ไม่มีเพิ่มเติม
5	23	$1 \leq N \leq 500,000$	$0 \leq M \leq 100,000$	ไม่มีเพิ่มเติม

## เกรดเดอร์ตัวอย่าง

เกรดเดอร์ตัวอย่างจะอ่านข้อมูลนำเข้าจากไฟล์ `horses.in` ในรูปแบบดังนี้:

- บรรทัดที่ 1:  $N$
- บรรทัดที่ 2:  $X[0] \dots X[N-1]$
- บรรทัดที่ 3:  $Y[0] \dots Y[N-1]$
- บรรทัดที่ 4:  $M$
- บรรทัดที่ 5, ...,  $M+4$ : เลขสามตัว `type pos val` (`type=1` เป็น `updateX` และ `type=2` เป็น `updateY`).

เกรดเดอร์ตัวอย่างจะพิมพ์ค่าคืนกลับ (return value) ของ `init` ตามด้วยค่าคืนกลับของทุกๆ การเรียก `updateX` และ `updateY`