

## เกมดันเจี้ยน

โรเบิร์ตกำลังออกแบบเกมคอมพิวเตอร์เกมใหม่ เกมนี้จะมีผู้กล้าหนึ่งคน ศัตรู  $n$  คน และดันเจี้ยน  $n + 1$  แห่ง ศัตรูจะมีหมายเลขกำกับตั้งแต่ 0 ถึง  $n - 1$  ส่วนดันเจี้ยนจะมีหมายเลขกำกับตั้งแต่ 0 ถึง  $n$  ศัตรู  $i$  (โดยที่  $0 \leq i \leq n - 1$ ) อยู่ที่ดันเจี้ยน  $i$  และมีพลัง  $s[i]$  ไม่มีศัตรูใด ๆ อยู่ในดันเจี้ยน  $n$

ผู้กล้าเริ่มเกมโดยเดินทางเข้าสู่ดันเจี้ยน  $x$  และมีพลัง  $z$  เมื่อใดก็ตามที่ผู้กล้าเข้าสู่ดันเจี้ยน  $i$  (โดยที่  $0 \leq i \leq n - 1$ ) เขาจะเผชิญหน้ากับศัตรู  $i$  และเหตุการณ์ใดเหตุการณ์หนึ่งต่อไปนี้จะเกิดขึ้น:

- ถ้าพลังของผู้กล้า มากกว่าหรือเท่ากับ พลังของศัตรู (ซึ่งก็คือ  $s[i]$ ) ผู้กล้าจะเป็นฝ่ายชนะ เหตุการณ์นี้จะทำให้ผู้กล้ามีพลังเพิ่มขึ้นจากเดิม  $s[i]$  (โดยที่  $s[i] \geq 1$ ) ในกรณีนี้ ผู้กล้าจะเข้าสู่ดันเจี้ยน  $w[i]$  เป็นลำดับถัดไป (โดยที่  $w[i] > i$ )
- มิฉะนั้น ผู้กล้าจะเป็นฝ่ายแพ้ เหตุการณ์นี้จะทำให้ผู้กล้ามีพลังเพิ่มขึ้นจากเดิม  $p[i]$  (โดยที่  $p[i] \geq 1$ ) ในกรณีนี้ ผู้กล้าจะเข้าสู่ดันเจี้ยน  $l[i]$  เป็นลำดับถัดไป

ทั้งนี้  $p[i]$  อาจจะน้อยกว่า เท่ากับ หรือมากกว่า  $s[i]$  ก็ได้ นอกจากนี้  $l[i]$  ก็อาจจะน้อยกว่า เท่ากับ หรือมากกว่า  $i$  ก็ได้ ไม่ว่าผลการเผชิญหน้าจะเป็นเช่นไร ศัตรูนั้นจะยังคงอยู่ในดันเจี้ยน  $i$  และมีพลัง  $s[i]$  เท่าเดิม

เกมนี้จบลงเมื่อผู้กล้าเข้าสู่ดันเจี้ยน  $n$  เราสามารถแสดงได้ว่า เกมนี้จะจบลงหลังจากเกิดการเผชิญหน้าเป็นจำนวนครั้งจำกัด (finite) ไม่ว่าผู้กล้าจะเริ่มต้นที่ดันเจี้ยนใด ด้วยพลังเท่าใดก็ตาม

โรเบิร์ตขอให้คุณช่วยทดสอบเกมของเขาโดยการจำลองการเล่นเกม  $q$  ครั้ง สำหรับการจำลองการเล่นเกมแต่ละครั้ง โรเบิร์ตจะกำหนดดันเจี้ยนเริ่มต้น  $x$  และพลังเริ่มต้น  $z$  มาให้ หน้าที่ของคุณคือให้หาว่า สำหรับการจำลองการเล่นเกมแต่ละครั้ง ผู้กล้าจะมีพลังเท่าใดเมื่อเกมจบลง

## รายละเอียดการเขียนโปรแกรม

คุณจะต้องเขียนฟังก์ชันต่อไปนี้:

```
void init(int n, int[] s, int[] p, int[] w, int[] l)
```

- $n$ : จำนวนศัตรู
- $s, p, w, l$ : อาร์เรย์ความยาว  $n$  โดยสำหรับ  $0 \leq i \leq n - 1$ :
  - $s[i]$  คือพลังของศัตรู  $i$  ซึ่งค่านี้ ยังเป็นปริมาณพลังที่ผู้กล้าจะได้เพิ่ม หลังจากเอาชนะศัตรู  $i$  ด้วย
  - $p[i]$  คือพลังที่ผู้กล้าจะได้เพิ่ม หลังจากพ่ายแพ้ต่อศัตรู  $i$
  - $w[i]$  คือดันเจี้ยนที่ผู้กล้าจะเข้าไป หลังจากเอาชนะศัตรู  $i$
  - $l[i]$  คือดันเจี้ยนที่ผู้กล้าจะเข้าไป หลังจากพ่ายแพ้ต่อศัตรู  $i$
- ฟังก์ชันนี้จะถูกเรียกเพียงครั้งเดียว ก่อนที่จะมีการเรียกฟังก์ชัน `simulate` (ดูด้านล่าง)

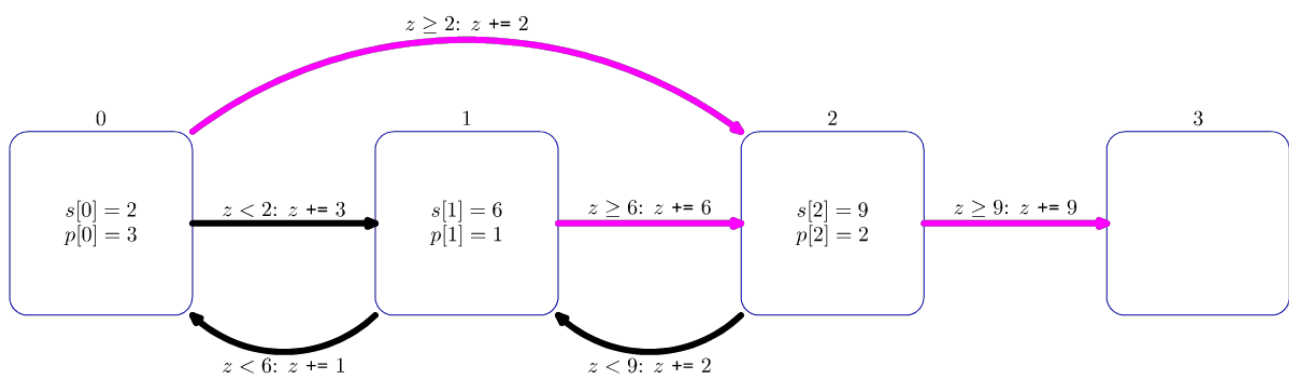
```
int64 simulate(int x, int z)
```

- $x$ : ดันเจี้ยนที่ผู้กล้าเข้าไปเป็นอันดับแรก
- $z$ : พลังตั้งต้นของผู้กล้า
- ฟังก์ชันนี้ต้องคืนค่าพลังของผู้กล้าเมื่อจบเกม หากผู้กล้าเริ่มเกมโดยเข้าสู่ดันเจี้ยน  $x$  และมีพลัง  $z$
- ฟังก์ชันนี้จะถูกเรียกทั้งสิ้น  $q$  ครั้ง

## ตัวอย่าง

พิจารณาการเรียกฟังก์ชันต่อไปนี้:

```
init(3, [2, 6, 9], [3, 1, 2], [2, 2, 3], [1, 0, 1])
```



แผนภาพด้านบนใช้อธิบายประกอบการเรียกฟังก์ชันนี้ ที่เหลื่อมจตุรัสแต่ละช่องแสดงแต่ละดันเจี้ยน สำหรับดันเจี้ยน 0, 1 และ 2 ค่าของ  $s[i]$  และ  $p[i]$  ถูกเขียนกำกับไว้ในสี่เหลียมจตุรัสนั้น ๆ ลูกศรสีม่วงแสดงการเคลื่อนที่ของผู้กล้าเมื่อได้รับชัยชนะจากการเผชิญหน้า ในขณะที่ลูกศรสีดำแสดงการเคลื่อนที่ของผู้กล้าเมื่อพ่ายแพ้

สมมุติว่าเกรดเดอร์เรียก `simulate(0, 1)`.

เกมจะดำเนินไปดังนี้:

ดันเจี้ยน	พลังของผู้กล้าก่อนการเผชิญหน้า	ผลการเผชิญหน้า
0	1	แพ้
1	4	แพ้
0	5	ชนะ
2	7	แพ้
1	9	ชนะ
2	15	ชนะ
3	24	เกมจบลง

ดังนั้น การเรียกฟังก์ชันครั้งนี้ต้องคืนค่า 24

สมมุติว่าเกรดเดอร์เรียก `simulate(2, 3)`

เกมจะดำเนินไปดังนี้:

ดัชนี	พลังของผู้กล้าก่อนการเผชิญหน้า	ผลการเผชิญหน้า
2	3	แพ้
1	5	แพ้
0	6	ชนะ
2	8	แพ้
1	10	ชนะ
2	16	ชนะ
3	25	เกมจบลง

ดังนั้น การเรียกฟังก์ชันครั้งนี้ต้องคืนค่า 25

## ข้อจำกัด

- $1 \leq n \leq 400\,000$
- $1 \leq q \leq 50\,000$
- $1 \leq s[i], p[i] \leq 10^7$  (สำหรับทุก  $0 \leq i \leq n - 1$ )
- $0 \leq l[i], w[i] \leq n$  (สำหรับทุก  $0 \leq i \leq n - 1$ )
- $w[i] > i$  (สำหรับทุก  $0 \leq i \leq n - 1$ )
- $0 \leq x \leq n - 1$
- $1 \leq z \leq 10^7$

## ปัญหาย่อย

1. (11 คะแนน)  $n \leq 50\,000$ ,  $q \leq 100$ ,  $s[i], p[i] \leq 10\,000$  (สำหรับทุก  $0 \leq i \leq n - 1$ )
2. (26 คะแนน)  $s[i] = p[i]$  (สำหรับทุก  $0 \leq i \leq n - 1$ )
3. (13 คะแนน)  $n \leq 50\,000$  และศัตรูทั้งหมดมีพลังเท่ากัน นั่นคือ  $s[i] = s[j]$  สำหรับทุก  $0 \leq i, j \leq n - 1$
4. (12 คะแนน)  $n \leq 50\,000$  และค่าของ  $s[i]$  เป็นค่าที่แตกต่างกันได้ไม่เกิน 5 ค่า
5. (27 คะแนน)  $n \leq 50\,000$
6. (11 คะแนน) ไม่มีข้อจำกัดเพิ่มเติม

## เกรดเดอร์ตัวอย่าง

เกรดเดอร์ตัวอย่างอ่านข้อมูลนำเข้าในรูปแบบต่อไปนี้:

- บรรทัดที่ 1:  $n\ q$
- บรรทัดที่ 2:  $s[0]\ s[1]\ \dots\ s[n - 1]$

- บรรทัดที่ 3:  $p[0] \ p[1] \ \dots \ p[n-1]$
- บรรทัดที่ 4:  $w[0] \ w[1] \ \dots \ w[n-1]$
- บรรทัดที่ 5:  $l[0] \ l[1] \ \dots \ l[n-1]$
- บรรทัดที่  $6 + i$  (สำหรับ  $0 \leq i \leq q-1$ ):  $x \ z$  สำหรับการเรียก simulate ครั้งที่  $i$

เกรดเดอร์ตัวอย่างจะพิมพ์คำตอบของคุณในรูปแบบต่อไปนี้:

- บรรทัดที่  $1 + i$  (สำหรับ  $0 \leq i \leq q-1$ ): ค่าที่การเรียก simulate ครั้งที่  $i$  ส่งคืนมา