



ผู้จัดการ (manager)

มีพนักงานทำงานนั่งติดกันเป็นแถวยาว N คน ($1 \leq N \leq 50\,000$) เรียกเป็นพนักงาน $0, 1, \dots, N-1$ ตามลำดับ พนักงาน i ทำงานกับผลิตภัณฑ์หมายเลข $A[i]$

ในโลกของการบริหารจัดการ พนักงานทุกคนที่เราสนใจจะต้องมีผู้จัดการ เพื่อความสะดวกผู้จัดการหนึ่งคนจะดูแลได้เฉพาะพนักงานที่อยู่ติดกันเท่านั้น นอกจากนี้ เนื่องจากผู้จัดการจะต้องคอยสังเกตการทำงาน ดังนั้นจำนวนผลิตภัณฑ์ที่พนักงานทำงานด้วยไม่ควรจะแตกต่างกันมากเกินไป

คุณจะได้รับคำถาม Q คำถาม ที่ระบุจำนวนเต็ม L, R , และ X ที่มีคุณจะต้องหาจำนวนผู้จัดการน้อยที่สุดที่

- สามารถดูแลพนักงานคนที่ L ถึง R (รวม L และ R ด้วย)
- ผู้จัดการแต่ละคนดูแลเฉพาะพนักงานหมายเลขติดกัน
- จำนวนผลิตภัณฑ์ที่พนักงานดูแลจะแตกต่างกันไม่เกิน X หมายเลข

พิจารณาตัวอย่างต่อไปนี้ที่ $N = 8$ และหมายเลขผลิตภัณฑ์ที่พนักงานแต่ละคนทำงานด้วยตามลำดับเป็นดังนี้

1, 20, 1, 40, 200, 1, 300, 20

สังเกตว่ามีผลิตภัณฑ์แตกต่างกัน 5 แบบ

สมมติว่า $Q = 3$ จะมีคำถาม 3 คำถาม ดังนี้

- ถ้าคำถามแรกถามโดยระบุ $L = 0, R = 7, X = 4$ คำตอบคือ 2 โดยรูปแบบหนึ่งที่เป็นไปได้คือผู้จัดการคนแรกดูแลพนักงานคนที่ $0 - 3$ และผู้จัดการคนที่ 2 ดูแลพนักงานคนที่ $4 - 7$
- ถ้าคำถามที่สองถามโดยระบุ $L = 2, R = 5, X = 3$ คำตอบคือ 1 เพราะว่าผู้จัดการคนเดียวสามารถดูแลพนักงานคนที่ $3 - 6$ ได้เลย เพราะว่าหมายเลขผลิตภัณฑ์ในช่วงดังกล่าวมีแค่ 3 แบบ คือ 1, 40, 200
- ถ้าคำถามที่สามถามโดยระบุ $L = 2, R = 7, X = 2$ คำตอบคือ 3 โดยผู้จัดการคนที่ 1 ดูแลพนักงานคนที่ $2 - 3$ ผู้จัดการคนที่ 2 ดูแลพนักงานคนที่ $4 - 5$ และผู้จัดการคนที่ 3 ดูแลพนักงานคนที่ $6 - 7$

รายละเอียดการเขียนโปรแกรม

คุณจะต้องเขียนฟังก์ชันสองฟังก์ชันด้านล่าง

```
void initialize(int N, int Q, vector<int> A)
```

- ฟังก์ชันจะถูกเรียกหนึ่งครั้ง
- อาร์เรย์ A ระบุความสูงของกล่อง กล่าวคือสำหรับ $0 \leq i \leq N-1$, $A[i]$ คือความสูงของกล่องของพนักงานคนที่ i

```
int min_managers(int L, int R, int X)
```

- ฟังก์ชันนี้จะถูกเรียก Q ครั้ง
- แต่ละครั้งให้คืนค่าจำนวนผู้จัดการน้อยที่สุดที่ตรงตามเงื่อนไข

เงื่อนไข

- $2 \leq N \leq 50\,000$
- $1 \leq Q \leq 50\,000$
- $1 \leq A[i] \leq 1\,000\,000\,000$ สำหรับทุก ๆ i ที่ $0 \leq i < N$
- $0 \leq L \leq R < N$
- $1 \leq X \leq 50\,000$

ปัญหาย่อย

1. (8 คะแนน) $N, Q \leq 1\,000$
2. (11 คะแนน) $L = 0, R = N - 1$ และมีค่า $A[i]$ ที่ต่างกันไม่เกิน 10 ค่า
3. (15 คะแนน) X เท่ากันในทุกคำถาม
4. (28 คะแนน) $L = 0$ และ $R = N - 1$
5. (38 คะแนน) ไม่มีเงื่อนไขเพิ่มเติม

ตัวอย่าง

เกรตเตอร์จะเรียกฟังก์ชัน `initialize` ดังนี้

```
initialize(8, 3, [1, 20, 1, 40, 200, 1, 300, 20])
```

เนื่องจาก $Q = 3$ จะมีคำถามตามมาอีก 3 คำถาม คำถามแรก เกรตเตอร์จะเรียก

```
min_managers(0, 7, 4)
```

ฟังก์ชันที่ทำงานถูกต้องจะคืนค่า 2 จากนั้นเกรตเตอร์จะเรียก

```
min_managers(2, 5, 3)
```

ฟังก์ชันที่ทำงานถูกต้องจะคืนค่า 1 จากนั้นเกรตเตอร์จะเรียก

```
min_managers(2, 7, 2)
```

ฟังก์ชันที่ทำงานถูกต้องจะคืนค่า 3

เกรตเตอร์ตัวอย่าง

เกรตเตอร์ตัวอย่างจะอ่านข้อมูลนำเข้าในรูปแบบดังนี้:

- บรรทัดที่ 1: $N \ Q$
- บรรทัดที่ 2: $A[0] \ A[1] \ A[2] \ \dots \ A[N-1]$
- บรรทัดที่ $3 + j$ ถึง $3 + Q - 1$: $L \ R \ X$

เกรตเตอร์ตัวอย่างจะพิมพ์ค่าที่คืนจากฟังก์ชัน `min_managers`

ขีดจำกัด

- Time limit: 7 seconds
- Memory limit: 1536 MB (1.5 GB)