



สับสวิตช์

ในขณะที่ทุกคนกำลังสอบแข่งเขียนโค้ดอยู่นั้น จู่ ๆ ไฟในห้องสอบก็ดับลง! (เป็นไปได้ว่าอาจารย์คุมสอบลืมจ่ายค่าไฟ) คุณซึ่งเป็นโปรแกรมเมอร์และช่างไฟที่เก่งกาจจึงมีหน้าที่ในการแก้ปัญหาด้วยการลงมือสับสวิตช์ไฟทั้งห้องให้กลับมาสว่างอีกครั้ง

สวิตช์ไฟในห้องจะเรียงกันเป็นแถบความยาว N ช่อง ระบุด้วยหมายเลข 0 ถึง $N - 1$ แต่ละช่องจะมีค่าประจำสวิตช์ $A[i]$

เริ่มต้นทุกช่องมีสถานะเป็น **ปิด** การสับสวิตช์ในช่องใด ๆ หนึ่งครั้งจะเปลี่ยนสถานะสวิตช์ในช่องดังกล่าวจาก **ปิด** → **เปิด** หรือจาก **เปิด** → **ปิด** คุณต้องการสับสวิตช์ให้ทุกช่องมีสถานะเป็น **เปิด** โดยมีเงื่อนไขการสับสวิตช์แต่ละรูปแบบอยู่ว่า

- การสับสวิตช์ในช่องที่ i จะใช้พลังงาน $A[i]$ หน่วย
- คุณสามารถเลือกสับสวิตช์เป็นช่วงจากช่วงที่กำหนดให้ M ช่วง โดยช่วงที่ j จะเปลี่ยนสถานะสวิตช์ตั้งแต่ตำแหน่ง $L[j]$ ถึง $R[j]$ และใช้พลังงาน $C[j]$ หน่วย
- ในตอนเริ่มต้นก่อนจะสับสวิตช์ในรูปแบบที่ 1 หรือ 2 คุณสามารถเลือกเปลี่ยนสถานะสวิตช์บนช่วงต่อเนื่องที่อาจซ้อนทับกันได้ โดยเปลี่ยนได้ไม่เกิน K ช่วง (อาจเปลี่ยนไม่ครบ K ช่วงหรือไม่เปลี่ยนเลยก็ได้) โดยในขั้นตอนนี้จะใช้พลังงาน $|A[i]^2 - A[j]^2|$ หน่วย ในการเปลี่ยนสถานะสวิตช์ตั้งแต่ช่องที่ i ถึงช่องที่ j ($i \leq j$) โดยอาจเลือกตำแหน่ง $i = j$ ได้

หน้าที่หลัก ๆ ของคุณก็มีเพียงแค่การเปิดไฟให้ครบทุกดวง แต่เนื่องจากคุณเป็นโปรแกรมเมอร์ คุณจึงต้องการหาค่าพลังงานที่น้อยที่สุดที่ต้องใช้ในการทำให้สวิตช์ทุกช่องมีสถานะเป็น **เปิด**

รายละเอียดการเขียนโปรแกรม

คุณจะต้องเขียนฟังก์ชันต่อไปนี้

```
long long minimum_energy(int N, int M, int K,
                          vector<int> A,
                          vector<int> L, vector<int> R, vector<int> C)
```

- ฟังก์ชันนี้จะถูกเรียกใช้โดยเกรดเดอร์เพียงครั้งเดียว

ขอบเขต

- $1 \leq N \leq 50\,000$
- $0 \leq M \leq 50\,000$
- $0 \leq K \leq 20$

- $1 \leq A[i] \leq 100\,000$ สำหรับ $0 \leq i \leq N - 1$
- $0 \leq L[j] \leq R[j] \leq N - 1$ สำหรับ $0 \leq j \leq M - 1$
- $1 \leq C[j] \leq 10^9$ สำหรับ $0 \leq j \leq M - 1$

ปัญหาย่อย

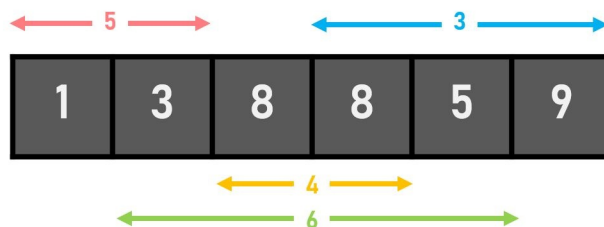
1. (3 คะแนน) $M = 0$ และ $K = 0$
2. (9 คะแนน) $R[j] \leq L[j + 1]$ และ $K = 0$
3. (21 คะแนน) $K = 0$
4. (19 คะแนน) $N, M \leq 1\,000$
5. (8 คะแนน) $A[i] \leq 500$
6. (18 คะแนน) $A[i] < A[i + 1]$
7. (22 คะแนน) ไม่มีเงื่อนไขเพิ่มเติม

ตัวอย่าง

พิจารณาการเรียกใช้ฟังก์ชันข้างต้น

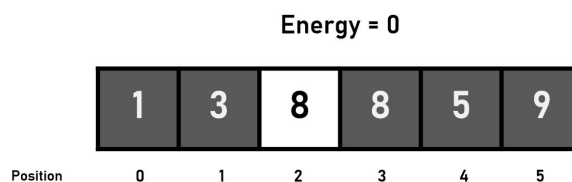
```
minimum_energy(6, 4, 1,
                [1, 3, 8, 8, 5, 9],
                [0, 1, 2, 3], [1, 4, 3, 5], [5, 6, 4, 3])
```

ในการเรียกใช้ครั้งนี้จะมีสวิตช์ 6 ช่อง มีค่าประจำสวิตช์เป็น $[1, 3, 8, 8, 5, 9]$

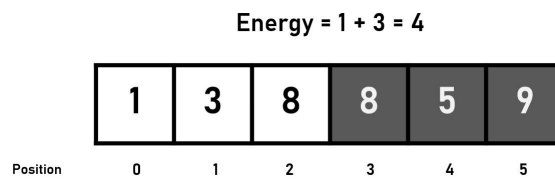


สังเกตว่าวิธีการเปลี่ยนสถานะสวิตช์ให้ทุกช่องเป็นสถานะเปิดที่ใช้พลังงานน้อยที่สุดจะทำได้โดย

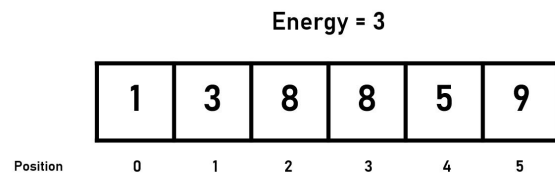
1. เปลี่ยนสถานะสวิตช์รูปแบบที่ 3 จากตำแหน่งที่ 2 ถึงตำแหน่งที่ 2 ใช้พลังงาน $|8^2 - 8^2| = 0$ หน่วย



2. เปลี่ยนสถานะสวิตช์รูปแบบที่ 1 บนตำแหน่งที่ 0 และ 1 ใช้พลังงาน $1 + 3 = 4$ หน่วย



3. เปลี่ยนสถานะสวิตช์รูปแบบที่ 2 บนช่วง $[3, 5]$ ใช้พลังงาน 3 หน่วย



รวมแล้วจึงใช้พลังงานเท่ากับ $0 + 4 + 3 = 7$ หน่วย ซึ่งเป็นวิธีที่ใช้พลังงานน้อยที่สุดเท่าที่เป็นไปได้ ฟังก์ชันนี้จึงควรส่งคืน 7

เกรตเตอร์ตัวอย่าง

เกรตเตอร์ตัวอย่างจะอ่านข้อมูลนำเข้าดังนี้

- บรรทัดที่ 1: $N \ M \ K$
- บรรทัดที่ 2: $A[0] \ A[1] \ A[2] \ \dots \ A[N-1]$
- บรรทัดที่ 3 ถึง $3 + M - 1$: $L[j] \ R[j] \ C[j]$ ในบรรทัดที่ $3 + j$

เกรตเตอร์ตัวอย่างจะส่งออกค่าที่ได้รับจาก `minimum_energy`

ข้อจำกัด

- Time limit: 1 second
- Memory limit: 512 MB