



โรบินฮู้ด (robinhood)

บ้านจำนวน N หลังเรียงต่อกัน บ้านที่ i สำหรับ $0 \leq i < N$ มีเงิน $A[i]$ บาท ($A[i]$ เป็นจำนวนเต็มคู่ ที่ไม่เป็นลบ) บ้านแต่ละหลังจะมีสถานะเป็นคณจหรือเป็นคณรย เมื่อตอนเริ่มต้นมีแค่บ้าน 0 เท่านั้นที่เป็นคณจ ระหว่างการทำงาน อาจจะมีบ้านหลังอื่น ๆ ที่เปลี่ยนสถานะเป็นคณจได้ แต่จะไม่มีใครกลับไปรยได้อีก สถานะคณจหรือรยไม่ขึ้นกับเงินที่บ้านนั้นมี

ตลอดการทำงาน จะมีเหตุการณ์เกิดขึ้น M ครั้ง

ในเหตุการณ์แรก วันหนึ่ง วิญญาณโรบินฮู้ดเข้าสิงเจ้าของบ้านที่ k (ซึ่งจะเรียกต่อไปว่าเป็นโรบินฮู้ด) โรบินฮู้ดนั้น ปล้นคณรย เพื่อมาช่วยคณจเลยจะทำให้ดำเนินการดังนี้

- โรบินฮู้ดจะเลือกบ้านคณจ สมมติว่าเป็นบ้านที่ x (อาจจะเป็นบ้านที่เอง k ด้วยก็ได้ ถ้าบ้าน k เป็นบ้านคณจแล้ว)
- โรบินฮู้ดจะเลือกบ้านที่อยู่ระหว่างบ้านที่ x และ k (รวม x กับ k ด้วย) ที่มีเงินมากที่สุด สมมติว่าเป็นบ้านที่ j และจะไปขโมยเงินจากบ้านดังกล่าวมา $A[j]$ บาท ก่อนจะหนีไปซ่อนที่บ้านหลังที่ x
- โรบินฮู้ดจะแบ่งเงินให้กับบ้านที่ x เพื่อเป็นค่าซ่อนตัวเท่ากับ 50% ของเงินที่บ้านนั้นมี นั่นคือ $A[x]/2$ บาท
- สุดท้ายโรบินฮู้ดจะปลอมตัวกลับบ้านตนเอง ด้วยเงินที่เหลือเท่ากับ $A[j] - A[x]/2$ บาท

เนื่องจากโรบินฮู้ดเป็นโจรคุณธรรมเขาจะอยากได้เงินติดตัวกลับไปน้อยที่สุด ให้อาว่าโรบินฮู้ดจะได้เงินกลับไปน้อยที่สุดกี่บาทในวันนั้น เมื่อสิ้นสุดกิจกรรมการขโมยนี้ ในวันถัดไปให้สมมติว่าเงินของทุกบ้านกลับมาเท่าเดิม

อีกเหตุการณ์หนึ่งที่เป็นไปได้ก็คือบางบ้านเปลี่ยนสถานะจากคณรยไปเป็นคณจ

พิจารณาตัวอย่างต่อไปนี้ สมมติว่า $N = 7$ และ $M = 5$ โดยที่เงินของแต่ละบ้านเป็นดังนี้

บ้านหลังที่ (i)	0	1	2	3	4	5	6
$A[i]$	2	400	2	10	200	0	20

เหตุการณ์ที่ 1: โรบินฮู้ดเข้าสิงบ้านที่ 2 เนื่องจากบ้านคณจมีแค่หลังเดียวคือบ้านที่ 0 โรบินฮู้ดจะเลือกบ้านที่มีเงินมากที่สุดจากบ้านที่ 0 ถึงบ้านที่ 2 ได้เป็นบ้านที่ 1 เขาจะปล้นบ้านดังกล่าว ได้เงินมา $A[1] = 400$ บาท นำไปจ่ายให้บ้าน 0 เท่ากับ $A[0]/2 = 2/2 = 1$ บาท และจะได้เงินเท่ากับ 399 บาท

เหตุการณ์ที่ 2: บ้าน 5 เปลี่ยนสถานะเป็นคณจ

เหตุการณ์ที่ 3: โรบินฮู้ดเข้าสิงบ้านที่ 3 เนื่องจากบ้านคณจมีบ้านหลังที่ 0 และ 5 เขาจะสามารถเลือกได้ว่าจะออกจากบ้าน 3 แล้วไปซ่อนตัวที่บ้าน 0 หรือ 5 ก็ได้ เขาเลือกซ่อนที่บ้านที่ 5 เพราะจะทำให้ได้เงินสุดท้ายเท่ากับ $200 - 0/2 = 200$ บาท (ปล้นบ้านที่ 4) ถ้าเขาไปซ่อนที่บ้าน 0 เขาจะได้เงิน $400 - 1/2 = 399$ ซึ่งมากกว่า 200

เหตุการณ์ที่ 4: บ้านที่ 6 เปลี่ยนสถานะเป็นคนจน

เหตุการณ์ที่ 5: โรบินฮู้ดเข้าสิงบ้านที่ 4 เนื่องจากบ้านคนจนมีบ้านหลังที่ 0, 5, และ 6 เขาจะสามารถเลือกได้ว่าจะออกจากบ้าน 4 ไปซ่อนตัวที่บ้าน 0, 5, หรือ 6 เขาเลือกปล้นบ้านที่ 4 (บ้านตัวเอง!) แล้วไปซ่อนที่บ้านที่ 6 ทำให้เขาได้เงินน้อยที่สุดคือ $200 - 20/2 = 190$ บาท

รายละเอียดการเขียนโปรแกรม

คุณจะต้องเขียนฟังก์ชันสามฟังก์ชันต่อไปนี้

```
void initialize(vector<int> A, int M)
```

- ฟังก์ชันนี้จะถูกเรียกใช้งาน 1 ครั้ง โดยจะส่งอาร์เรย์ A ขนาด N ให้กับฟังก์ชันและระบุจำนวนครั้ง M ของเหตุการณ์

จากนั้นเกรตเตอร์จะเรียกฟังก์ชันด้านล่างฟังก์ชันใดฟังก์ชันหนึ่ง (add หรือ ask) ในลำดับอย่างไรก็ได้ จำนวน M ครั้ง

```
void add(int y)
```

- เกรตเตอร์จะเรียกฟังก์ชันนี้เพื่อระบุว่าบ้าน y เปลี่ยนสถานะเป็นคนจน

```
int ask(int k)
```

- เกรตเตอร์จะเรียกฟังก์ชันนี้เพื่อระบุเหตุการณ์ที่โรบินฮู้ดเข้าสิงบ้านที่ k คุณจะต้องคืนค่าเงินที่น้อยที่สุดที่โรบินฮู้ดจะทำได้ในกรณีนี้

เงื่อนไข

- $1 \leq N \leq 500\,000$
- $1 \leq M \leq 500\,000$
- $0 \leq A[i] \leq 100\,000\,000$ และ $A[i]$ เป็นจำนวนคู่ไม่น้อยกว่า 0

ปัญหาย่อย

- (6 points) $N \leq 5\,000$, $M \leq 5\,000$
- (19 points) $N \leq 50\,000$, $M \leq 50\,000$
- (8 points) $k = n - 1$ และ $N \leq 200\,000$, $M \leq 200\,000$
- (9 points) เรียก add ก่อน ask และ $N \leq 200\,000$, $M \leq 200\,000$
- (12 points) $N \leq 200\,000$, $M \leq 200\,000$
- (9 points) $k = n - 1$
- (12 points) เรียก add ก่อน ask

8. (25 points) ไม่มีเงื่อนไขเพิ่มเติมอื่น ๆ จากโจทย์

ตัวอย่าง

จากตัวอย่างข้างต้นเกรดเดอร์จะเรียก

```
initialize([2, 400, 2, 10, 200, 0, 20], 5)
```

จากนั้นจะเรียกฟังก์ชันอีก 5 ครั้งดังนี้

สำหรับเหตุการณ์ที่ 1 เกรดเดอร์จะเรียก

```
ask(2)
```

ซึ่งจะต้องคืนค่า 399 จากนั้นจะเรียก

```
add(5)
```

เพื่อบอกว่าบ้าน 5 เปลี่ยนสถานะเป็นคนจน

สำหรับเหตุการณ์ที่ 3 เกรดเดอร์จะเรียก

```
ask(3)
```

ซึ่งจะต้องคืนค่า 200 จากนั้นสำหรับเหตุการณ์ที่ 4 จะเรียก

```
add(6)
```

เพื่อบอกว่าบ้าน 6 เปลี่ยนสถานะเป็นคนจน

สุดท้ายสำหรับเหตุการณ์ที่ 5 จะเรียก

```
ask(4)
```

ซึ่งจะต้องคืนค่า 190

เกรดเดอร์ตัวอย่าง

เกรดเดอร์ตัวอย่างจะอ่านข้อมูลนำเข้าดังนี้

- บรรทัดที่ 1: $N \ M$
- บรรทัดที่ 2: $A[0] \ A[1] \ \dots \ A[N-1]$
- บรรทัดที่ $3 + i$ ถึง $3 + M - 1$: ระบุเหตุการณ์ M เหตุการณ์ในรูปแบบดังนี้

- เหตุการณ์เปลี่ยนสภาพบ้าน y : 1 y
- เหตุการณ์โรบินฮู้ดเข้าสิงบ้าน k : 2 k

เกรตเตอร์ตัวอย่างจะพิมพ์ค่าที่ได้จากการเรียกฟังก์ชัน `ask`

ขีดจำกัด

- Time limit: 1 seconds
- Memory limit: 1024 MB