



งานเลี้ยงบนเกาะ

ในจังหวัดแห่งหนึ่ง มีเกาะประหลาดแสนพิศวงอยู่ทั้งหมด N เกาะ โดยเกาะที่ i จะมีค่าความมินงอยู่ $A[i]$ ดิกรีนอกจากนี้ยังมีสะพานเชื่อมแต่ละเกาะอยู่ทั้งหมด $N - 1$ แห่ง โดยสะพานแห่งที่ j จะเชื่อมเกาะที่ $u[j]$ และ $v[j]$ เข้าด้วยกัน โดยรับประกันว่าทุกเกาะสามารถเดินทางไปหากันได้โดยใช้สะพานและจะมีเพียงหนึ่งเส้นทางเท่านั้น และยังรับประกันอีกว่าเกาะใด ๆ จะถูกเชื่อมด้วยสะพานไม่เกิน 20 แห่งเท่านั้น

นิยามค่าความเมา d ในการเดินทางจากเกาะที่ a ใด ๆ ไปเกาะที่ b ใด ๆ คือ การนำค่าความมินง A ของทุกเกาะบนเส้นทางจากเกาะที่ a ไปยังเกาะที่ b ทั้งหมดมา XOR กัน (รวมเกาะที่ a และ b ด้วย)

ผู้ว่าราชการจังหวัดของจังหวัดนี้ ได้เล็งเห็นว่าผู้คนบนเกาะเกิดความเครียด เนื่องจากกลัวว่าพรรคการเมืองที่ตนได้ทำการเลือกไว้จะไม่ชนะการเลือกตั้งในครั้งนี้ ท่านผู้ว่าจึงได้ประกาศจัดงานเลี้ยง (ปาร์ตี้สังฆินเมา) ขึ้นมาทั้งสิ้น T คีน โดยในแต่ละคีน ท่านผู้ว่าจะชวนคนจากเกาะที่ L ถึงเกาะที่ R ให้มาร่วมงานเลี้ยงในคีนนั้น โดยงานเลี้ยงจะถูกจัดขึ้นบนเกาะที่ X (X ไม่จำเป็นต้องอยู่ในช่วงเกาะที่ L ถึงเกาะที่ R) เนื่องจากผู้คนบนเกาะต้องการคลายความเครียดที่เกิดขึ้นของตน จึงตัดสินใจเดินทางไปร่วมงานเลี้ยงอย่างแน่นอน โดยจะนิยามค่าความมินเมา D ของการจัดงานเลี้ยงคือ การนำค่าความเมา d ของการเดินทางจากเกาะที่ Y ใด ๆ ในช่วง L ถึง R ไปยังเกาะที่ X ทั้งหมดมา XOR กัน

คุณซึ่งเครียดกับผลลัพธ์ที่จะเกิดขึ้นของการเลือกตั้งเช่นกัน แต่ก็ไม่อยากที่จะเอาตัวเองเข้าไปสู่ความมินเมา จึงเลือกที่จะคลายเครียดด้วยการหาว่าค่าความมินเมา D ที่เกิดขึ้นในแต่ละคีนนั้นมีค่าเท่าไร

Implementation Details

คุณจะต้องเขียนฟังก์ชันสองฟังก์ชันต่อไปนี้

```
void init(int N, int T, vector<int> A, vector<int> u, vector<int> v)
```

- ฟังก์ชันนี้จะถูกเรียกใช้โดยเกรดเดอร์เพียงครั้งเดียว
- N : จำนวนเกาะทั้งหมด
- T : จำนวนคีนที่ถูกจัดงานเลี้ยง
- A : ค่าความมินงของเกาะที่ i สำหรับทุก $0 \leq i \leq N - 1$
- u, v : คู่ของเกาะที่ถูกเชื่อมด้วยสะพานแห่งที่ j สำหรับทุก $0 \leq j \leq N - 2$
- ฟังก์ชันนี้ไม่มีการคืนค่า

```
long long find_drunkeness(int L, int R, int X)
```

- ฟังก์ชันนี้จะถูกเรียกใช้โดยเกรดเดอร์ทั้งหมด T ครั้ง
- L, R : ช่วงของเกาะที่ถูกชวนให้มาร่วมงานเลี้ยง

- X : เกาะที่เป็นเจ้าภาพในการจัดงานเลี้ยง
- ฟังก์ชันนี้จะต้องคืนค่าเป็นจำนวนเต็ม แทนค่าความมินเมาที่เกิดขึ้นในคืนนั้น

Constraints

- $2 \leq N \leq 100\,000$
- $1 \leq T \leq 100\,000$
- $0 \leq A[i] \leq 2^{60} - 1$ สำหรับทุก $0 \leq i \leq N - 1$
- $0 \leq u[j], v[j] \leq N - 1$ และ $u[j] \neq v[j]$ สำหรับทุก $0 \leq j \leq N - 2$
- $0 \leq L \leq R \leq N - 1$ และ $0 \leq X \leq N - 1$
- เกาะใด ๆ จะถูกเชื่อมด้วยสะพานไม่เกิน 20 แห่งเท่านั้น

Subtasks

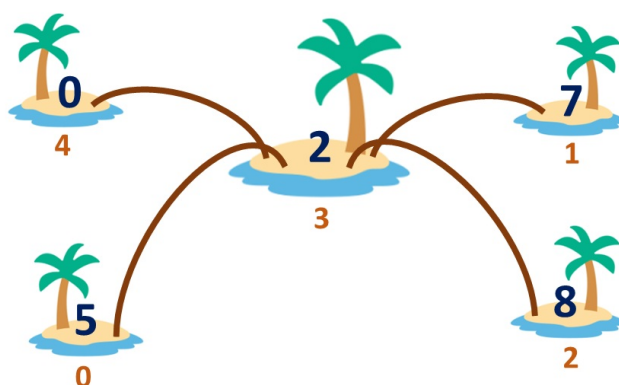
1. (4 คะแนน) $N, T \leq 3\,000$
2. (5 คะแนน) มี X ที่เป็นไปได้เพียงหนึ่งค่า
3. (20 คะแนน) เกาะใด ๆ จะถูกเชื่อมด้วยสะพานไม่เกิน 2 แห่งเท่านั้น
4. (16 คะแนน) มี L และ R ที่เป็นไปได้เพียงหนึ่งค่า
5. (28 คะแนน) $L = 0$ และ R ในแต่ละคืนจะมีค่ามากกว่าหรือเท่ากับ R ในคืนก่อนหน้า
6. (27 คะแนน) ไม่มีเงื่อนไขเพิ่มเติม

Examples

พิจารณาการเรียกใช้ฟังก์ชันต่อไปนี้

```
init(5, 4, [5, 7, 8, 2, 0], [4, 3, 3, 0], [3, 1, 2, 3])
```

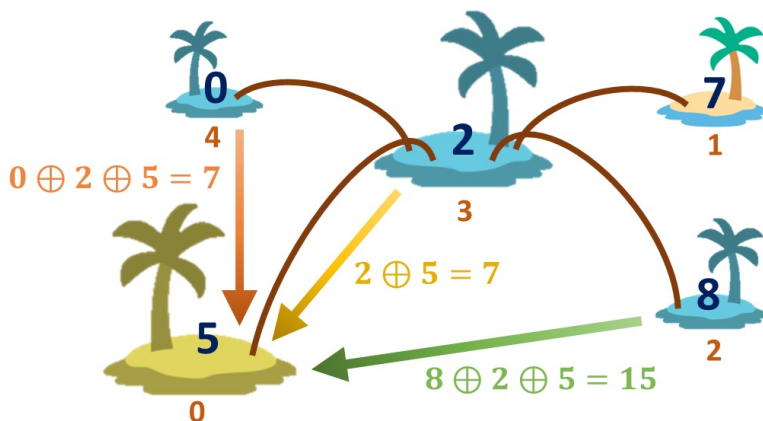
สำหรับเหตุการณ์นี้ จะมีเกาะอยู่ทั้งหมด 5 เกาะ ซึ่งมีค่าความมินเมาเป็น $[5, 7, 8, 2, 0]$ และมีสะพานเชื่อมต่อเกาะเข้าหากันดังรูป



จากนั้นจะมีการจัดงานเลี้ยงขึ้นทั้งหมด 4 คืน ดังนี้

```
find_drunkenness(2, 4, 0)
```

สำหรับงานเลี้ยงในคืนที่ 1 งานเลี้ยงถูกจัดขึ้นบนเกาะที่ 0 โดยได้ชวนคนจากเกาะที่ 2 ถึงเกาะที่ 4 ให้มาร่วมงานเลี้ยง ซึ่งจะทำให้เกิดค่าความเมาจากทั้ง 3 เส้นทาง ดังนี้



จากค่าความเมาของทั้ง 3 เส้นทาง จะได้ว่าค่าความมึนเมาของงานเลี้ยงในคืนนี้จะมีค่าเท่ากับ $15 \oplus 7 \oplus 7 = 15$ ซึ่งฟังก์ชันจะต้องคืนค่า 15 โดย \oplus แทนการดำเนินการ XOR

```
find_drunkenness(0, 1, 3)
```

สำหรับงานเลี้ยงในคืนที่ 2 ฟังก์ชันจะต้องคืนค่า 2 เนื่องจากค่าความเมาของเส้นทางจากเกาะที่ 0 มายังเกาะที่ 3 คือ $5 \oplus 2 = 7$ และค่าความเมาจากเกาะที่ 1 มาเกาะที่ 3 คือ $7 \oplus 2 = 5$ ทำให้ได้ค่าความมึนเมาของงานเลี้ยงคือ $7 \oplus 5 = 2$

```
find_drunkenness(2, 3, 1)
```

สำหรับงานเลี้ยงในคืนที่ 3 ฟังก์ชันจะต้องคืนค่า 8

```
find_drunkenness(4, 4, 2)
```

สำหรับงานเลี้ยงในคืนที่ 4 ฟังก์ชันจะต้องคืนค่า 10

Sample Grader

เกรดเดอร์ตัวอย่างจะอ่านข้อมูลนำเข้าดังนี้

- บรรทัดที่ 1: $N \ T$
- บรรทัดที่ 2: $A[0] \ A[1] \ A[2] \ \dots \ A[N-1]$

- $N - 1$ บรรทัดถัดมา: $u[j] \ v[j]$
- T บรรทัดถัดมา: $L \ R \ X$

เกรตเตอร์ตัวอย่างจะส่งออกมาที่ได้รับจาก `find_drunkenness` ทั้งหมด T ครั้ง

Limits

- Time limit: 2 seconds
- Memory limit: 256 MB