



ห่างเพียงใด

อาณาจักรข้ามจักรวาลแห่งหนึ่งมีสถานีอวกาศเรียงต่อกันเป็นเส้นตรงจำนวน N สถานี เรียกเป็นสถานี $0, 1, \dots, N - 1$ สถานีที่ 0 ตั้งอยู่ที่โลกซึ่งเป็นศูนย์กลางของจักรวาล

สถานีเหล่านี้แม้จะเรียงเป็นเส้นตรงแต่ก็อยู่ห่างกันมาก การเดินทางระหว่างสถานีจะใช้เวลาวิ่งผ่านทางรูนอนที่เดินทางได้ทางเดียวเท่านั้น สำหรับ $0 \leq i < N - 1$ จะมีรูนอนจากสถานี i ไปยังสถานี $i + 1$ นอกจากนี้อาจจะมีการเพิ่มรูนอนเข้ามาได้อีก แต่ทิศทางการเคลื่อนที่จะวิ่งออกจากสถานี 0 เสมอ นั่นคือทุก ๆ รูนอนที่เพิ่มเข้ามา จะทำให้เดินทางจากสถานี u ไปยังสถานี v ที่ $u < v$ เสมอ

อาณาจักรดังกล่าวแม้ว่าจะยิ่งใหญ่ แต่ก็ยังมีบางช่วงเวลา que สถานีอวกาศบางแห่งประท้วงไม่ยอมให้ยานเดินทางผ่าน วันหนึ่งผู้อำนวยการที่อยู่ at สถานีที่ 0 ต้องการเดินทางไปยังสถานี t แต่ทราบว่าทุก ๆ สถานีตั้งแต่สถานี u ถึงสถานี v มีการประท้วง (รวมสถานี u และ v ด้วย) ผู้อำนวยการอยากทราบว่ายังสามารถเดินทางจากสถานี 0 ไปยังสถานี t โดยไม่ผ่านสถานีที่มีการประท้วงเลยได้หรือไม่

คุณต้องเขียนโปรแกรมเพื่อรองรับเหตุการณ์ดังนี้

- เพิ่มรูนอน `add_arc(u, v)`: เพิ่มรูนอนที่เดินทางเดียวจากสถานี u ไปสถานี v (รับประกันว่า $u < v$)
- ตอบคำถาม `is_reachable(u, v, t)`: ตอบคำถามว่า จากสถานี 0 เดินทางไปยังสถานี t โดยไม่ผ่านสถานีตั้งแต่ u ไปจนถึง v ได้หรือไม่?

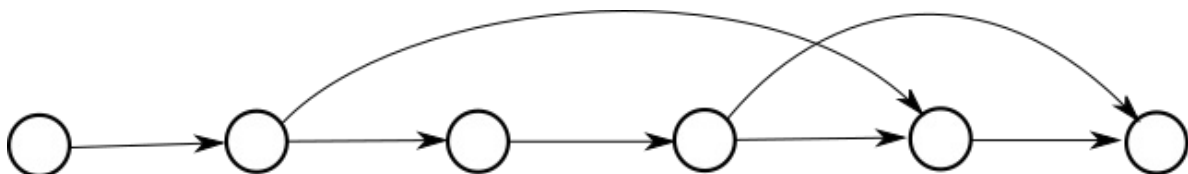
เหตุการณ์มีทั้งสิ้น Q เหตุการณ์

พิจารณาตัวอย่างต่อไปนี้ สมมติให้ $N = 6$ ถ้ายังไม่มี การเพิ่มรูนอนใด ๆ ถ้ามีการถามว่า

- `is_reachable(4, 5, 1)` จะตอบว่า true เพราะว่าสามารถเดินทางได้ผ่านทางรูนอนเชื่อมสถานี 0 กับ 1 โดยไม่ต้องผ่านสถานี 4 ถึง 5
- `is_reachable(3, 3, 5)` จะพบว่าจากสถานี 0 ไม่สามารถเดินทางไปยังสถานี 5 โดยไม่ผ่านสถานี 3 ได้ ดังนั้นต้องตอบ false

ถ้ามีการเพิ่มรูนอนสองรูดังนี้

- `add_arc(1, 4)` และ `add_arc(3, 5)` ผลลัพธ์จะแสดงดังรูปด้านล่าง



ถ้ามีคำถามดังนี้

- `is_reachable(2, 3, 4)` จะตอบว่า `true` เนื่องจากสามารถใช้รูหนอนเชื่อมระหว่างสถานี 1 กับ 4 ในการกระโดดข้ามสถานี 2 กับ 3 ได้
- `is_reachable(2, 4, 5)` จะตอบว่า `false` เนื่องจากไม่มีทางเดินทางจากสถานี 0 ไปยังสถานี 5 ได้โดยไม่ผ่านสถานี 2 ถึง 4 แม้จะมีรูหนอนที่เพิ่มมาแล้ว

รายละเอียดการเขียนโปรแกรม

คุณจะต้องเขียนฟังก์ชันสามฟังก์ชันต่อไปนี้

```
void initialize(int N)
```

- ฟังก์ชันนี้จะถูกเรียกหนึ่งครั้ง

หลังจากเรียก `initialize` เกรดเดอร์จะเรียกฟังก์ชันด้านล่าง รวมแล้วไม่เกิน Q ครั้ง

คุณต้องเขียนฟังก์ชันด้านล่างเพื่อรองรับการเพิ่มรูหนอน

```
void add_arc(int u, int v)
```

- ระบุว่ามีการเพิ่มเส้นเชื่อม u ไป v
- รับประกันว่า $u < v$
- $0 \leq u < v \leq N - 1$

คุณต้องเขียนฟังก์ชันต่อไปนี้เพื่อตอบคำถาม

```
bool is_reachable(int u, int v, int t)
```

- ถามว่ามีเส้นทางจาก 0 ไป t ที่ไม่ผ่านจุดยอด w ใดๆ ที่ $u \leq w \leq v$
- รับประกันว่า $u \leq v$ เป็นไปได้ที่ $u = v$
- ฟังก์ชันต้องคืนค่า `true` และ `false`
- $0 \leq t \leq N - 1$

ขอบเขต

- $1 \leq N \leq 100\,000$
- $1 \leq Q \leq 100\,000$
- $0 \leq u \leq v \leq N - 1$
- $0 \leq t \leq N - 1$

ปัญหาย่อย

1. (5 คะแนน) $N \leq 1\,000$, $Q \leq 1\,000$

2. (17 คะแนน) ไม่มีเส้นเชื่อมที่เพิ่มไปสองเส้นใด ๆ ที่ตัดกัน นั่นคือสำหรับเส้นเชื่อมสองเส้น (a, b) กับ (c, d) ที่ $a < c$ จะไม่มีกรณีที่ $c < b < d$; และทุก ๆ คำถาม `is_reachable` จะตามหลังการเรียก `add_arc` ทั้งหมด
3. (21 คะแนน) ไม่มีเส้นเชื่อมที่เพิ่มไปสองเส้นใด ๆ ที่ตัดกัน (ตามนิยามข้างต้น)
4. (25 คะแนน) ทุก ๆ คำถาม `is_reachable` จะตามหลังการเรียก `add_arc` ทั้งหมด
5. (32 คะแนน) ไม่มีเงื่อนไขเพิ่มเติม

ตัวอย่าง

จากตัวอย่างข้างต้น การเรียกฟังก์ชัน `initialize` เริ่มต้นจะเป็นดังนี้

```
initialize(6)
```

หลังจากนั้นการทำงานจะเป็นดังนี้ (ในกรณีนี้ $Q = 6$)

เหตุการณ์	การเรียกฟังก์ชัน	ผลลัพธ์	หมายเหตุ
1	<code>is_reachable(4, 5, 1)</code>	true	
2	<code>is_reachable(3, 3, 5)</code>	false	
3	<code>add_arc(1, 4)</code>		เพิ่มรูลนอนจากสถานี 1 ไป 4
4	<code>add_arc(3, 5)</code>		เพิ่มรูลนอนจากสถานี 3 ไป 5
5	<code>is_reachable(2, 3, 4)</code>	true	
6	<code>is_reachable(2, 4, 5)</code>	false	

เกรดเดอร์ตัวอย่าง

เกรดเดอร์ตัวอย่างอ่านข้อมูลดังนี้

- บรรทัดที่ 1: N, Q
- บรรทัดที่ $2, \dots, Q + 1$: เป็นข้อใดข้อหนึ่งต่อไปนี้
 - 1 $u\ v$: เพิ่มรูลนอนจากสถานี u ไป v
 - 2 $u\ v\ t$: ถามคำถามว่าจะสามารถเดินทางจาก 0 ไป t โดยที่ไม่ผ่านสถานีตั้งแต่ u ถึง v ได้หรือไม่

เกรดเดอร์จะพิมพ์ค่าที่คืนจากฟังก์ชัน `is_reachable` ถ้าเป็น true จะพิมพ์ 1 ถ้าเป็น false จะพิมพ์ 0

ข้อจำกัด

- Time limit: 1.5 second
- Memory limit: 512 MB

