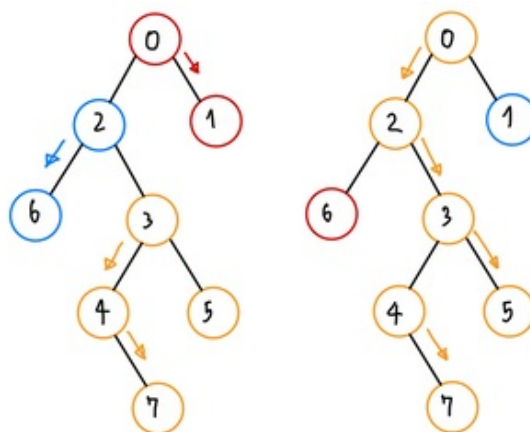


ลวดลาย (Pattern)

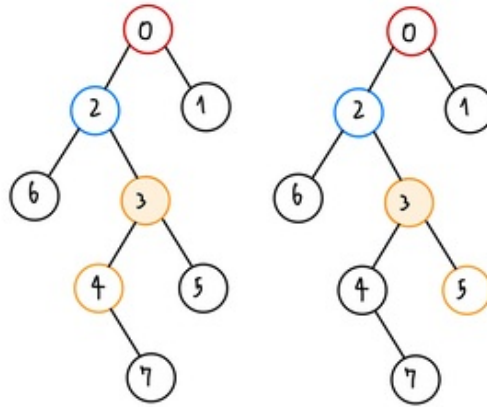
ในอุตสาหกรรมแห่งการทอผ้านั้นมีการแข่งขันทางด้านผลผลิตของผ้าเป็นอย่างมาก ปัจจัยหลักที่ส่งผลต่อการผลิตคือจำนวนสีบนผ้า กรรมวิธีที่ย้อมสีบนผ้า และการย้อมสีบนผ้าในลักษณะพิเศษซึ่งก่อให้เกิดลวดลาย สำหรับการผลิตผ้าคุณภาพยอดเยี่ยมนั้นจะมีขั้นตอนพิเศษเพิ่มขึ้นดังนี้

1. ผ้าจะถูกจัดให้จุดสำคัญของผ้ามีทั้งหมด N จุดโดยมีหมายเลขตั้งแต่ 0 ถึง $N - 1$ มีเส้นทางระหว่างจุดสำคัญให้แบคทีเรียผ่านได้ทั้งหมด $N - 1$ ทาง และรับประกันว่าทุกจุดสำคัญสองจุดใดๆ จะมีเส้นทางไปหากันได้
2. จะมีการจัดผ้าให้จุดสำคัญที่ 0 เป็นจุดสูงสุด และจะมีการนิยามปลายผ้าเป็นจุดสำคัญที่ไม่มีลูกของมัน
3. แบคทีเรียจะมีทั้งหมด M ประเภทซึ่งแบคทีเรียที่ i จะให้สีที่ i สำหรับ $0 \leq i \leq M - 1$
4. จะมีการปล่อยแบคทีเรียสำหรับย้อมผ้าบนจุดสำคัญต่างๆ สำหรับแบคทีเรียที่ยังไม่เคยกระจายที่จุดสำคัญใดๆ จะกระจายไปที่ลูกที่ยังไม่มีแบคทีเรียเพียงจุดสำคัญเดียวเท่านั้น และจะกระจายต่อไปเรื่อยๆ จนกว่าจะถึงปลายผ้า
5. หลังจากแบคทีเรียกระจายครบทุกจุดสำคัญแล้ว แบคทีเรียจะทำการย้อมสีจุดสำคัญนั้นๆ
6. ทุกจุดสำคัญต้องถูกย้อมสี
7. เนื่องจากผ้าคุณภาพยอดเยี่ยมนั้นมีลวดลายเฉพาะตัวจึงทำให้มีการบังคับสีทั้งหมด P จุดสำคัญโดยจะมีการบอกว่าที่จุดสำคัญ A_i จะต้องเป็นสี B_i
8. จำนวนแบคทีเรียที่ปล่อยบนจุดสำคัญมีจำนวนน้อยที่สุดเท่าที่จะทำได้เพื่อให้สามารถย้อมสีทุกจุดสำคัญได้ (อาจจะไม่มีวิธีที่สอดคล้องขั้นตอนที่ 7)

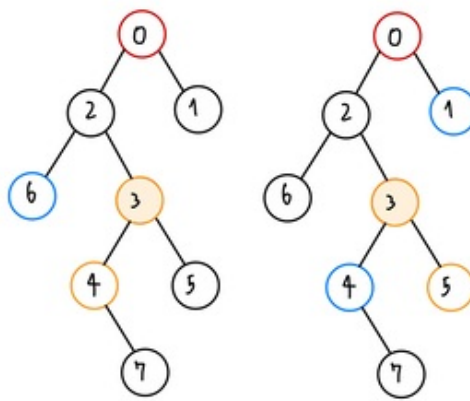
ทางโรงงานจึงต้องการทราบจำนวนผลลัพธ์สุดท้ายของการย้อมผ้า เนื่องจากคำตอบอาจจะมีค่ามากเกินไปจึงขอให้ตอบเศษที่เหลือจากการหารด้วย $10^9 + 7$ แทน



รูปด้านบนนี้แสดงวิธีการปล่อยแบคทีเรียที่ถูกต้อง โดยรูปแรกจะปล่อยแบคทีเรียทั้งหมด 4 ครั้งโดยจะปล่อยที่จุด 0, 3, 5, 2 และรูปที่สองจะปล่อยแบคทีเรียทั้งหมด 4 ครั้งโดยจะปล่อยที่จุด 0, 1, 4, 6



ให้จุดสำคัญ 3 เป็นจุดที่มีการถูกกำหนดสี จะพบว่าทั้งสองรูปจะเป็นวิธีการปล่อยแบคที่เรียกที่ถูกต้องและให้ผลลัพธ์สุดท้ายแบบเดียวกัน



จากรูปด้านบนจะเป็นตัวอย่างการปล่อยแบคที่เรียกที่ไม่ถูกต้อง รูปด้านซ้ายผิดเงื่อนไขเนื่องจากแบคที่เรียกที่จุดสำคัญ 0 ต้องกระจายไปทั้งสองจุดสำคัญคือ 1, 2 เพื่อให้ทุกจุดสำคัญถูกย้อมสี รูปด้านขวาผิดเงื่อนไขเนื่องจากมีการวางแบคที่เรียกมากกว่าจำนวนแบคที่เรียกขั้นต่ำที่ต้องวาง

รายละเอียดการเขียนโปรแกรม

คุณจะต้องเขียนฟังก์ชันดังต่อไปนี้

```
int total_pattern(int N, int M, vector<vector<int> > Path,
                 vector<vector<int> > P)
```

- ฟังก์ชันนี้จะถูกเรียกเพียงครั้งเดียวสำหรับปัญหาย่อย
- เวกเตอร์ Path จะมีขนาด $N - 1$ และประกอบด้วยเวกเตอร์ที่มีขนาด 2 โดย $Path[i][0]$ จะมีทางเชื่อมไปหา $Path[i][1]$ สำหรับ $0 \leq i \leq N - 2$
- เวกเตอร์ P จะมีขนาด P และประกอบด้วยเวกเตอร์ที่มีขนาด 2 โดย $P[i][0]$ คือ A_i และ $P[i][1]$ คือ B_i สำหรับ $0 \leq i \leq P - 1$
- ฟังก์ชันนี้ต้องคืนค่าเป็นจำนวนเต็มหนึ่งจำนวนคือจำนวนผลลัพธ์สุดท้ายของการย้อมผ้า

ขอบเขต

- $1 \leq N, M \leq 200\,000$
- $0 \leq P \leq 5$
- $0 \leq A_i \leq N - 1$
- $0 \leq B_i \leq M - 1$
- ที่จุดสำคัญใดๆ จะมีการถูกกำหนดสีไม่เกิน 1 ครั้ง

ปัญหาย่อย

1. (11 คะแนน) จุดสำคัญแต่ละจุดจะมีเส้นทางไม่เกิน 2 เส้น
2. (11 คะแนน) $M \leq 2, P = 0$
3. (12 คะแนน) $M \leq 3, P = 0$
4. (9 คะแนน) $M \leq 3$
5. (15 คะแนน) $N, M \leq 1\,000$
6. (24 คะแนน) $P = 0$
7. (18 คะแนน) ไม่มีเงื่อนไขเพิ่มเติม

ตัวอย่าง

ตัวอย่างที่ 1

```
total_pattern(3, 2, [[0, 1], [0, 2]], [])
```

จะคืนค่า 6

ตัวอย่างที่ 2

```
total_pattern(3, 2, [[0, 1], [0, 2]], [[1, 0]])
```

จะคืนค่า 3

ตัวอย่างที่ 3

```
total_pattern(3, 2, [[0, 1], [0, 2]], [[1, 0], [2, 1]])
```

จะคืนค่า 2

ตัวอย่างที่ 4

```
total_pattern(5, 10, [[0, 1], [1, 2], [2, 3], [3, 4]], [[1, 7], [3, 6  
]])
```

จะคืนค่า 0

เกรตเตอร์ตัวอย่าง

เกรตเตอร์ตัวอย่างจะมีการรับข้อมูลดังนี้

- บรรทัดที่ 1: $N \ M \ P$
- บรรทัดที่ 2 ถึง N : $Path[i][0] \ Path[i][1]$
- บรรทัดที่ $N + 1$ ถึง $N + P$: $A_i \ B_i$

ข้อจำกัด

- Time limit: 1 second
- Memory limit: 128 MB