



```
int count_roads(int N, int M, int K,  
               vector<vector<int>> S,  
               vector<pair<int,int>> R)
```

- ฟังก์ชันจะถูกเรียกหนึ่งครั้ง
- พารามิเตอร์  $N$  ระบุจำนวนเมืองและ  $M$  ระบุจำนวนถนนทั้งหมด และ  $K$  ระบุจำนวนถนนสายหลัก
- สำหรับ  $0 \leq j < K$  อาร์เรย์  $S[j]$  จะระบุรายการของเมืองในถนนสายหลักสายที่  $j$  ไล่ไปตามลำดับจากเมืองหลวง รายการนี้ไม่รวมเมืองหลวงหมายเลข 0
- สำหรับ  $0 \leq i < R.size()$  จะมีถนนเส้นตรงแบบทางเดียวจากเมือง  $R[i][0]$  ไปยังเมือง  $R[i][1]$
- ฟังก์ชันจะคืนค่าจำนวนถนนที่สามารถกลับทิศได้

## เงื่อนไข

- $1 \leq N \leq 100\,000$
- $1 \leq M \leq 300\,000$
- $1 \leq K \leq 10$

## ปัญหาย่อย

1. (5 points)  $K = 1$
2. (10 points)  $N \leq 300, M \leq 1000$
3. (16 points)  $K = 2$  ถนนทุกเส้นบนเส้นทางหลัก  $K$  เส้นสามารถกลับทิศได้
4. (26 points)  $K = 2$
5. (23 points)  $K = 3$
6. (20 points) ไม่มีเงื่อนไขอื่น

## ตัวอย่าง

สำหรับตัวอย่างด้านบน กริดเดอร์จะเรียกฟังก์ชัน

```
count_roads(8, 12, 2,  
            [[1, 7, 5, 6],  
             [2, 3, 4]],  
            [[0,7], [3,1], [3,5], [5,4], [2,4]])
```

ฟังก์ชันจะต้องคืนค่า 8

## กริดเดอร์ตัวอย่าง

กริดเดอร์ตัวอย่างจะอ่านข้อมูลนำเข้าในรูปแบบต่อไปนี้

- บรรทัดที่ 1:  $N \ M \ K$

- บรรทัดที่  $2 + j$  ถึง  $2 + K - 1$ :  $L[j]$   $S[j][0]$   $S[j][1]$   $S[j][2]$  ...  $S[j][L[j]]$  โดยที่  $L[j]$  เท่ากับจำนวนเมืองที่ไม่ใช่เมืองหลวงในทางสายหลักสายที่  $j$
- บรรทัดที่  $2 + i$  ถึง  $2 + M - 1$ :  $R[i][0]$   $R[i][1]$

เกรเดอร์ตัวอย่างจะพิมพ์ผลลัพธ์ที่คืนจากฟังก์ชัน `count_roads`

## ข้อจำกัด

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