

전파 방해 (jamming)

종영국의 군대는 1번 부대부터 N 번 부대까지 N 개의 부대로 이루어져 있다. 부대들은 전파 통신을 통해 서로에게 정보를 전송하는데, M 개의 양방향 통신이 존재하며, i 번째 통신은 A_i 번 부대와 B_i 번 부대 간의 통신이다($1 \leq i \leq M$). 사건이 발생했을 때 모든 부대들에 그 정보를 전송해야 하기 때문에 모든 부대들은 하나 이상의 통신을 통해 다른 모든 부대들에 정보를 전송할 수 있다. 또한 연결하는 부대 쌍이 같은 부대 쌍이 여럿이라면 비효율적이므로 각 전파 통신에 대해 같은 부대 쌍을 연결하는 다른 전파 쌍은 존재하지 않는다.

지민국의 왕 지민이는 종영국을 침략하기 위해 부대 간의 전파 통신을 방해해 정상적인 활동을 할 수 없게 하는 계획을 세웠다. 지민국은 총 $M - 4$ 개의 전파 방해 도구를 가지고 있어, $M - 4$ 개의 전파 통신을 무력화시킬 수 있다.

가능한 모든 통신이 무력화된 부대들은 활동을 멈추지만, 하나 이상의 통신이 여전히 가능한 부대들은 활동을 지속할 것이다. 지민이는 무력화 후 모든 ‘활동 가능한 부대’들이 하나 이상의 통신을 통해 다른 모든 ‘활동 가능한 부대’에 정보를 전송할 수 있다면, 완벽한 무력화에 실패했다고 판단한다. 그렇지 않은 경우는 완벽히 무력화에 성공한 것이다.

지민이를 위해 $M - 4$ 개의 전파 통신을 무력화해 종영국 군대를 완벽히 무력화하는 방법의 수를 구해주자.

구현 세부 사항

여러분은 아래 함수를 구현해야 한다.

```
int jamming(int N, int[] A, int[] B)
```

- 정수 N 은 종영국의 부대의 수 N 을 의미한다.
- 정수 배열 A 는 길이 M 의 수열로, $A[0], A[1], \dots, A[M-1]$ 에 A_1, \dots, A_M 이 저장되어 있다.
- 정수 배열 B 는 길이 M 의 수열로, $B[0], B[1], \dots, B[M-1]$ 에 B_1, \dots, B_M 이 저장되어 있다.
- 이 함수는 $M - 4$ 개의 전파 통신을 무력화해 종영국 군대를 완벽히 무력화하는 방법의 수를 $10^9 + 7$ 로 나눈 나머지를 반환해야 한다.

제약 조건

- $4 \leq N \leq 200\,000$
- $4 \leq M \leq 200\,000$
- $1 \leq A_i \leq N$ ($1 \leq i \leq M$)
- $1 \leq B_i \leq N$ ($1 \leq i \leq M$)
- $A_i \neq B_i$ ($1 \leq i \leq M$)
- 각 전파 통신에 대해, 연결하는 부대 쌍이 같은 다른 전파 통신은 존재하지 않는다.
- 모든 부대들이 하나 이상의 통신을 통해 다른 모든 부대들에 정보를 전송할 수 있다.

부분문제

1. (5점) $M \leq 100$
2. (7점) $M \leq 5\,000$
3. (13점) $M = N - 1$
4. (27점) 각 그룹마다 그룹 내의 두 부대를 연결하는 통신이 없도록 부대들을 두 그룹으로 나눌 수 있다.
5. (28점) 모든 부대에 대해, 해당 부대에서 x 개의 서로 다른 통신들을 이용해 $x - 1$ 개의 서로 다른 부대를 거쳐 다시 해당 부대로 정보를 전송할 수 있다면, $x \leq 3$ 을 만족한다.
6. (20점) 추가 제한 조건 없음.

예제

- $N = 5, M = 7, A = [1, 2, 3, 4, 3, 2, 5], B = [2, 3, 4, 1, 5, 4, 4]$. 그레이더는 다음의 함수를 호출한다.

`jamming(5, [1, 2, 3, 4, 3, 2, 5], [2, 3, 4, 1, 5, 4, 4]) = 2`

이 예제는 1, 2, 6번 부분문제의 조건을 만족한다.

- $N = 8, M = 10, A = [1, 1, 2, 2, 2, 6, 6, 6, 7, 7], B = [3, 8, 3, 4, 8, 4, 5, 8, 4, 5]$. 그레이더는 다음의 함수를 호출한다.

`jamming(8, [1, 1, 2, 2, 2, 6, 6, 6, 7, 7], [3, 8, 3, 4, 8, 4, 5, 8, 4, 5]) = 159`

이 예제는 1, 2, 4, 6번 부분문제의 조건을 만족한다.

- $N = 9, M = 11, A = [1, 2, 1, 2, 3, 5, 7, 7, 3, 5, 5], B = [2, 3, 3, 4, 5, 6, 9, 8, 6, 8, 7]$. 그레이더는 다음의 함수를 호출한다.

`jamming(9, [1, 2, 1, 2, 3, 5, 7, 7, 3, 5, 5], [2, 3, 3, 4, 5, 6, 9, 8, 6, 8, 7]) = 256`

이 예제는 1, 2, 5, 6번 부분문제의 조건을 만족한다.

- $N = 14, M = 13, A = [1, 1, 1, 1, 1, 2, 2, 3, 3, 4, 4, 5, 5], B = [2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14]$. 그레이더는 다음의 함수를 호출한다.

`jamming(14, [1, 1, 1, 1, 1, 2, 2, 3, 3, 4, 4, 5, 5], [2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14]) = 622`

이 예제는 모든 부분문제의 조건을 만족한다.

Sample grader

Sample grader는 아래와 같은 형식으로 입력을 받는다.

- Line 1: N M
- Line $1 + i$: A_i B_i ($1 \leq i \leq M$)