

학생들 (students)

IOI 대학교는 학생들을 이름이 아니라 시험 등수로 부르는 삭막한 경쟁 사회이다. 대학교에는 N 명의 학생이 있으며, 각 $0 \leq i \leq N - 1$ 에 대해 i 번 학생은 중간고사에서 $i + 1$ 등을 한 학생이다.

학생들은 기말고사를 대비하기 위해 자체적으로 멘토링 그룹을 꾸렸다. 각 멘토링 그룹은 두 정수 $0 \leq L \leq R \leq N - 1$ 로 나타낼 수 있는데, 번호가 L 이상 R 이하인 학생들을 **제외한** 모든 학생이 이 그룹에 속해 있다는 의미이다.

어느 날 아침, IOI 대학교의 모든 학생들에게 “멘토링 그룹이 1개 이상 존재한다” 라는 메일이 발송되었다. 이를 1일차 아침이라고 할 때, $d \geq 1$ 일차 저녁부터는 다음과 같은 상황이 진행된다.

- 어떤 학생이 d 일차 아침 이후에 자신을 배제한 멘토링 그룹이 존재한다는 사실을 추론하였다. 학생은 불공평함을 느끼고, d 일차 저녁이 끝나기 전에 민원을 접수한다. 민원이 접수되면, $d + 1$ 일차 아침이 오기 전 모든 멘토링 그룹의 활동이 종료된다.
- 어떤 학생도 자신을 배제한 멘토링 그룹이 존재한다는 사실을 추론하지 못하였다. 민원이 접수되지 않고, $d + 1$ 일차 아침에도 멘토링 그룹 활동이 지속된다. 이를 통해 모든 학생은 “아무도 d 일차에 민원을 접수하지 않았다” 라는 공통적인 정보를 인지한다.

이외에 학생들은 어떤 방식으로든 서로의 정보를 공유하지 않는다. 즉, 자신이 속한 멘토링 그룹과 각 그룹의 구성원, 민원 접수 여부만 알 수 있다. 학생은 자신이 가진 정보로 추론할 수 있는 명제를 항상 추론하며, 자신의 가진 정보에서 틀릴 가능성이 있는 명제를 추론하지 않는다.

당신은 IOI 대학교의 모든 학생과 친한 외부인으로, 모든 멘토링 그룹과 그 구성원을 알고 있다. M 개의 모든 멘토링 그룹에 대한 정보가 주어질 때, 민원이 접수되는 날짜 k 와, k 일차에 민원을 접수하는 학생의 번호를 모두 구하는 프로그램을 작성하여라. 영원히 민원이 접수되지 않는 경우에는 -1 을 반환하면 된다.

그룹의 수 M , 각 그룹의 구성원, 전체 문제 및 부분 문제의 제한, **모든 그룹이 특정 범위의 번호를 갖는 학생들을 제외한 꼴이라는 사실** 등 모든 멘토링 그룹에 대한 정보는 오직 외부인인 여러분만 알 수 있고, 각 학생은 전혀 알지 못하는 정보임에 유의하라.

함수 목록 및 정의

여러분은 아래 함수를 구현해야 한다.

```
pair<int, vector<int> > complaint(int N, vector<int> L, vector<int> R)
```

- L, R : 크기가 M 인 정수 배열. 모든 $0 \leq i \leq M - 1$ 에 대해, i 번 멘토링 그룹은 번호가 $L[i]$ 미만이거나 $R[i]$ 초과인 학생들로만 구성되어 있다.
- 이 함수는 정수 k 와 크기가 N 이하인 정수 배열 V 로 이루어진 순서쌍을 반환해야 한다. k 는 민원이 접수되는 날짜이고, V 는 k 일차에 민원을 접수하는 학생들의 번호를 **증가하는 순서대로** 담고 있어야 한다. 영원히 민원이 접수되지 않을 경우 k 는 -1 , V 는 빈 배열이 되어야 한다.

제출하는 소스 코드의 어느 부분에서도 입출력 함수를 실행해서는 안 된다.

제약 조건

- $1 \leq N \leq 250\,000$
- $1 \leq M \leq 250\,000$
- 모든 $0 \leq i \leq M - 1$ 에 대해 $0 \leq L[i] \leq R[i] \leq N - 1$

부분문제

1. (12점)

- $N, M \leq 10$

2. (6점)

- 모든 $0 \leq i \leq M - 1$ 에 대해 $L[i] = R[i]$.

3. (15점)

- $N, M \leq 2\,500$.

4. (10점)

- 모든 $0 \leq i, j \leq M - 1$ 에 대해 구간 $[L[i], R[i]]$ 와 $[L[j], R[j]]$ 는 서로소이거나 서로를 포함한다.
- 즉 $L[i] < L[j]$ 라면, $R[i] < L[j]$ 또는 $R[j] \leq R[i]$ 가 성립한다.

5. (18점)

- 모든 $0 \leq i \leq M - 2$ 에 대해 $L[i] < L[i + 1]$, $R[i] < R[i + 1]$ 이 성립한다.

6. (39점)

- 추가적인 제약 조건이 없다.

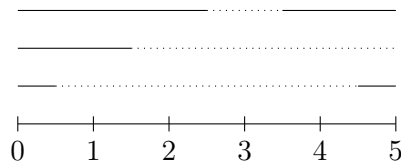
예제 1

$N = 6$, $M = 3$, $L = [1, 2, 3]$, $R = [4, 5, 3]$ 인 경우를 생각해 보자.

그레이더는 다음과 같이 함수를 호출한다.

```
complaint(6, [1, 2, 3], [4, 5, 3])
```

아래 그림의 실선은 각 멘토링 그룹이 포함하는 인원을 나타낸다.



3번 학생은 모든 멘토링 그룹에서 제외되었기 때문에, 첫날 바로 민원을 접수한다. 그 이외의 학생은 첫날 민원을 접수하지 않으므로, 함수는 $k = 1$ 과 $V = [3]$ 을 순서쌍으로 묶어 $(1, [3])$ 을 반환해야 한다.

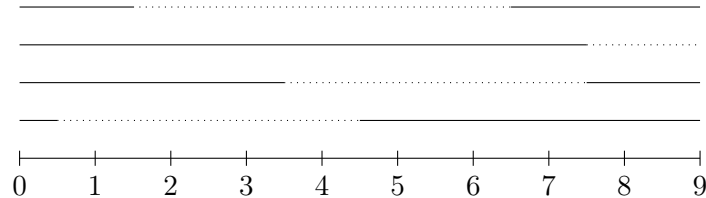
예제 2

$N = 10$, $M = 4$, $L = [1, 4, 8, 2]$, $R = [4, 7, 9, 6]$ 인 경우를 생각해 보자.

그레이더는 다음과 같이 함수를 호출한다.

```
complaint(10, [1, 4, 8, 2], [4, 7, 9, 6])
```

아래 그림의 실선은 각 멘토링 그룹이 포함하는 인원을 나타낸다.



함수는 $k = 2$ 과 $V = [4, 8, 9]$ 를 순서쌍으로 묶어 $(2, [4, 8, 9])$ 를 반환해야 한다.

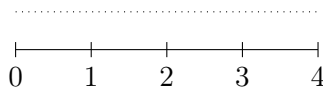
예제 3

$N = 5$, $M = 1$, $L = [0]$, $R = [4]$ 인 경우를 생각해 보자.

그레이더는 다음과 같이 함수를 호출한다.

```
complaint(5, [0], [4])
```

아래 그림의 실선은 각 멘토링 그룹이 포함하는 인원을 나타낸다. 이 멘토링 그룹은 아무 학생도 포함하고 있지 않다. (누가 이런 멘토링 그룹을 만들었을까?)



함수는 $k = 1$ 과 $V = [0, 1, 2, 3, 4]$ 를 순서쌍으로 묶어 $(1, [0, 1, 2, 3, 4])$ 를 반환해야 한다.

Sample grader

Sample grader는 아래와 같은 형식으로 입력을 받는다.

- Line 1: N M
- Line $2 + i$ ($0 \leq i \leq M - 1$): $L[i]$ $R[i]$

Sample grader는 다음을 출력한다.

- Line 1: complaint 함수가 반환한 값 k
- Line 2: complaint 함수가 반환한 값 V 의 원소들을 공백으로 구분하여 출력한다.

Sample grader는 실제 채점에서 사용하는 그레이더와 다를 수 있음에 유의하라.