

## 회의실

$K$ 개의 회의실이 있다.  $N$ 개의 회의가 이 회의실들을 이용하려고 한다. 각 회의는 1부터  $N$ 까지 번호가 붙어 있다. 회의  $i$ 는 시작 시각  $s_i$ , 끝나는 시각  $e_i$ , 위약금  $w_i$ 로 표현된다.

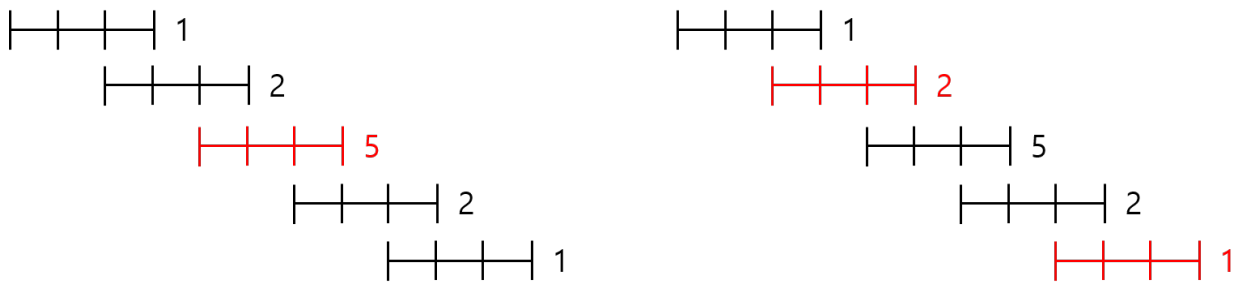
이 회의실들은 매우 특이한 규칙에 따라 운영된다. 회의  $i$ 와 회의  $j$ 는 다음 조건 중 적어도 하나를 만족하면 **관련있는 회의**라고 부른다.

1. 두 구간  $[s_i, e_i]$ 와  $[s_j, e_j]$ 의 공통된 구간이 존재할 때 (시작과 끝 시각만 겹치는 경우에도)  $i$ 와  $j$ 는 **관련있는 회의**이다.
2. 두 구간  $[s_i, e_i]$ 와  $[s_j, e_j]$ 의 공통된 구간이 존재하지 않지만, 다른 회의  $c$ 가  $i$ 와 관련있고  $[s_c, e_c]$ 와  $[s_j, e_j]$ 의 공통된 구간이 존재할 때 (시작과 끝 시각만 겹치는 경우에도)  $i$ 와  $j$ 는 **관련있는 회의**이다.

회의들이 취소될 수 있어서 위의 정의는 취소되지 않은 회의들만 가지고 판단한다. 즉, 원래 **관련있는 회의** 관계였던 두 회의가, 일부 회의의 취소로 인해 **관련있는 회의** 관계가 아니게 될 수 있다.

취소되지 않은 회의는 회의실을 하나씩 할당받아야 한다. **관련있는 회의**들은 모두가 반드시 **서로 다른** 회의실을 할당받아야 한다. 예를 들어 세 회의  $[1, 3]$ ,  $[3, 5]$ ,  $[5, 7]$ 는,  $[1, 3]$ 과  $[5, 7]$  사이에 공통된 구간이 없어도 불구하고, 두 개가 아닌 세 개의 회의실을 할당받아야 함에 유의하시오. 회의실이  $K$ 개밖에 없기 때문에 일부 회의들을 취소해야 할 수 있다. 회의  $i$ 를 취소하면 위약금  $w_i$ 를 내야 하므로, **가급적** 취소할 회의를 잘 골라서 지급할 위약금의 합을 **최소**로 하고 싶다.

다음 그림은 5개의 회의  $[1, 4]$ ,  $[3, 6]$ ,  $[5, 8]$ ,  $[7, 10]$ ,  $[9, 12]$ 가 있고, 각 회의의 위약금이 차례로 1, 2, 5, 2, 1인 경우를 보이고 있다. 회의실이 2개 있다고 하자. 왼쪽의 예는  $[5, 8]$ 을 취소해서 조건을 만족시키는 경우이며, 이 경우 위약금은 5이다. 오른쪽의 예는  $[3, 6]$ 과  $[9, 12]$ 를 취소해서 조건을 만족시키는 경우이며, 이 경우 위약금은 3이다. 모든 경우를 고려해보면 **최소**의 위약금 합은 3임을 알 수 있다.



## 함수 목록 및 정의

여러분은 아래 함수를 구현해야 한다.

```
long long int min_charge(int K, vector<int> S, vector<int> E, vector<int> W)
```

- 이 함수는 정확히 한 번 호출된다.
- $K$ 는 회의실의 개수이다.
- $S, E, W$ 의 길이는 모두  $N$ 이다.
- 회의  $i + 1$ 의 시작 시각이  $S[i]$ , 종료 시각이  $E[i]$ , 위약금이  $W[i]$ 이다( $0 \leq i \leq N - 1$ ).
- 이 함수는 주어진 회의실의 수와 회의 정보를 바탕으로, 조건을 만족하게 하기 위해서 취소해야 할 회의들 중 가장 위약금의 합이 적은 경우를 찾고 그때의 위약금의 합을 반환해야 한다.

제출하는 소스 코드의 어느 부분에서도 입출력 함수를 실행해서는 안 된다.

## 제약 조건

- $1 \leq K \leq N$
- $1 \leq N \leq 2500$
- $1 \leq s_i \leq e_i \leq 10^9$  ( $1 \leq i \leq N$ )
- $1 \leq w_i \leq 10^9$  ( $1 \leq i \leq N$ )

## 부분문제

- (10점)
  - $N \leq 16$
- (17점)
  - $K = 1$
- (32점)
  - 모든  $i$ 에 대해  $w_i = 1$
- (26점)
  - $N \leq 250$
- (65점)
  - 추가적인 제약 조건이 없다.

## 채점 기준

`min_charge` 함수가 반환한 위약금의 합이 정답과 일치해야 한다.

각 부분문제의 점수는 그 부분문제의 모든 데이터에 대한 점수 중 최솟값임에 유의하라.

## 예제

- $N = 5, K = 2, S = [1, 3, 5, 7, 9], E = [4, 6, 8, 10, 12], W = [1, 2, 5, 2, 1]$  인 경우를 생각해 보자.

그레이더는 다음 함수를 호출한다.

```
min_charge(2, [1, 3, 5, 7, 9], [4, 6, 8, 10, 12], [1, 2, 5, 2, 1])
```

본문의 예제 설명에 따라, `min_charge` 함수는 종료하면서 3을 반환해야 한다.

이 예제는 부분문제 2와 3의 조건을 만족하지 않음에 유의하라.

## Sample grader

Sample grader는 아래와 같은 형식으로 입력을 받는다.

- Line 1:  $N K$
- Line  $1 + i$  ( $1 \leq i \leq N$ ):  $S[i - 1] E[i - 1] W[i - 1]$

Sample grader는 아래와 같은 형식으로 출력한다.

- Line 1: 함수 `min_charge`가 반환한 값

Sample grader는 실제 채점에서 사용하는 그레이더와 다를 수 있음에 유의하라.