

## 날다람쥐 (squirrel)

2차원 평면 위에  $N$ 개의 기둥이 일렬로 놓여 있다. 기둥들에는 왼쪽에서 오른쪽으로 1번부터  $N$ 번까지의 자연수 번호가 붙어 있다.

$i$  ( $1 \leq i \leq N$ )번째 기둥의 바닥은 점  $(D_i, 0)$ 에 위치하고, 높이는  $H_i$ 이다. 따라서 이 기둥은 점  $(D_i, 0)$ 와  $(D_i, H_i)$ 를 잇는 선분이다. 또한,  $D_1 = 0$ 이다.

처음에 날다람쥐는 제일 왼쪽 기둥의 높이  $L$ 인 곳, 즉 점  $(0, L)$ 에 있다. 날다람쥐는 **모든** 기둥을 **왼쪽부터 순서대로** 거쳐서 제일 오른쪽 기둥의 높이  $R$ 인 곳, 즉 점  $(D_N, R)$ 에 가려고 한다.

날다람쥐가 한 기둥에서 다음 기둥으로 날아갈 때 오른쪽으로  $d$  ( $d \geq 0$ )만큼 움직이면 높이가  $d$ 만큼 감소한다. 다음 기둥에 도착하기 전에 땅에 닿으면 안 된다. 다음 기둥의 높이 0인 곳에 도착하는 것은 허용된다.

날다람쥐는 한 기둥에서 위로 기어오르거나 아래로 내려갈 수 있다. 기둥의 높이보다 더 높은 곳으로 오를 수는 없다.  $i$ 번째 기둥에서 위로  $h$  ( $h \geq 0$ )만큼 오르면  $W_i \times h$ 의 비용이 든다. 기둥에서 아래로 내려갈 때는 비용이 들지 않는다.

아래 **그림 1**은 날다람쥐가 이동하는 한 가지 예이다.

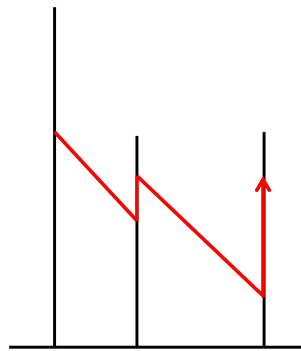


그림 1

**그림 2**의 왼쪽처럼 이동하는 것은 중간에 땅에 닿은 경우가 있어 허용되지 않는다. **그림 2**의 오른쪽처럼 이동하는 것은 기둥을 거치지 않은 경우가 있어 역시 허용되지 않는다.

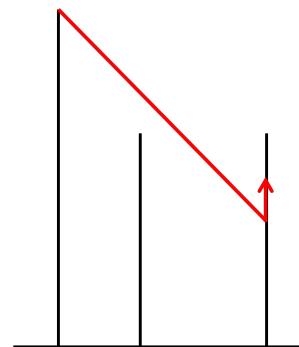
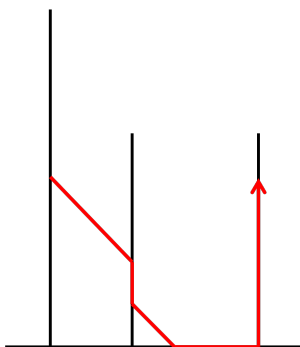


그림 2

가장 작은 총 비용으로 날다람쥐가 목표 위치에 도착할 수 있는 방법을 계산하라.

## 함수 목록 및 정의

여러분은 아래 함수를 구현해야 한다.

```
long long fly(vector<int> D, vector<int> H, vector<int> W, int L, int R)
```

- 이 함수는 단 한 번만 호출된다.
- 인자로 주어지는 3개 배열의 크기는 기둥의 개수  $N$ 이다.
- 인자로 주어지는 정수 배열  $D$ 의  $D[i]$ 는 1번째 기둥에서  $i + 1$ 번째 기둥까지의 거리  $D_{i+1}$ 이다.
- 인자로 주어지는 정수 배열  $H$ 의  $H[i]$ 는  $i + 1$ 번째 기둥의 높이  $H_{i+1}$ 이다.
- 인자로 주어지는 정수 배열  $W$ 의  $W[i]$ 는  $i + 1$ 번째 기둥에서 거리 1을 올라갈 때의 비용  $W_{i+1}$ 이다.
- 인자로 주어지는 정수  $L$ 은 1번째 기둥에서 날다람쥐가 최초에 위치한 높이이다.
- 인자로 주어지는 정수  $R$ 은  $N$ 번째 기둥에서 날다람쥐가 도착해야 하는 높이이다.
- 이 함수의 반환값은:
  - 규칙을 지키면서 최종 위치에 도착하는 방법이 있는 경우, 날다람쥐가 최종 위치에 도착하는 최소 비용이어야 한다. 제약 조건을 만족하는 입력 데이터에서의 최소 비용은 항상 정수임을 증명할 수 있다.
  - 규칙을 지키면서 최종 위치에 도착하는 방법이 없는 경우,  $-1$ 이어야 한다.

제출하는 소스 코드의 어느 부분에서도 입출력 함수를 실행해서는 안 된다.

## 제약 조건

- $2 \leq N \leq 500\,000$
- $0 = D_1 < D_2 < \dots < D_N \leq 10^9$
- $1 \leq H_i \leq 10^9$  ( $1 \leq i \leq N$ )
- $0 \leq W_i \leq 10^9$  ( $1 \leq i \leq N$ )
- $0 \leq L \leq H_1$
- $0 \leq R \leq H_N$

## 부분문제

1. (4점)

- $W_i = 0$  ( $1 \leq i \leq N$ )

2. (13점)

- $W_i = 1$  ( $1 \leq i \leq N$ )

3. (18점)

- $W_i \leq W_{i+1}$  ( $1 \leq i \leq N - 1$ )

4. (15점)

- $N \leq 500$
- $H_i \leq 500$  ( $1 \leq i \leq N$ )

5. (18점)

- $N \leq 5000$

6. (32점)

- 추가적인 제약 조건이 없다.

## 채점 기준

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

## 예제

- $N = 3$ 이고 1번 기둥에서 2번 기둥까지의 거리는 2, 1번 기둥에서 3번 기둥까지의 거리는 5, 기둥들의 높이는 왼쪽부터 각각 8, 5, 5, 기둥들의 가중치는 왼쪽부터 각각 3, 4, 6,  $L = 5$ ,  $R = 4$ 인 경우를 생각해 보자.

그레이더는 다음 함수를 호출한다.

```
fly([0, 2, 5], [8, 5, 5], [3, 4, 6], 5, 4) = 18
```

이 경우 1번 기둥에서 거리 2만큼 올라가고, 3번 기둥에서 거리 2만큼 올라가는 것이 최적이므로 함수는 18을 반환해야 한다.

이 예제는 부분문제 3, 4, 5, 6의 조건을 만족한다.

## Sample grader

Sample grader는 아래와 같은 형식으로 입력을 받는다.

- Line 1:  $N$
- Line  $1 + i$  ( $1 \leq i \leq N$ ):  $D_i H_i W_i$
- Line  $2 + N$ :  $L R$

Sample grader는 다음을 출력한다.

- Line 1: 함수 `f1y`가 반환한 값

Sample grader는 실제 채점에서 사용하는 그레이더와 다를 수 있음에 유의하라.