Flying Squirrel 2

There are a flying squirrel and N pillars on a 2-dimensional plane.

A point on the 2-dimensional plane can be represented as (x, y). Here, x denotes the horizontal position, and y denotes the height. The direction in which the horizontal position x increases is to the right, and the direction in which the height y increases is upwards.

The pillars are numbered from 0 to $N-1$ in order. The base of pillar i ($0 \le i \le N-1$) is located at the point $(i, 0)$, and its height is infinite. Therefore, pillar i is a ray starting from the point $(i, 0)$ and extending upwards. Each pillar is either red or blue. If $B[i] = 0$, pillar i is red, and if $B[i] = 1$, pillar i is blue.

Initially, the flying squirrel is at point $(0, 0)$. The flying squirrel wants to reach point (N, H). To do this, the flying squirrel moves in the following way.

In places without pillars, the flying squirrel flies to the right while maintaining its current height. Since the flying squirrel is very fast, the time taken in this case is considered to be 0.

Where there is a pillar, the flying squirrel can increase its height by 1 or do nothing. Specifically, at the location of pillar i ($0 \le i \le N-1$), the flying squirrel must perform one of the following actions:

- **Pass** the pillar. The flying squirrel's height remains unchanged, and it continues to fly to the right. The time taken is 0.
- **Climb** the pillar. This action is possible only when the pillar is red ($B[i] = 0$). The flying squirrel's height increases by 1 at the pillar, and then it flies to the right again. The time taken is $A[i]$.
- **Jump** from the pillar. This action is possible only when the pillar is blue ($B[i] = 1$). The flying squirrel's height increases by 1 at the pillar, and then it flies to the right again. The time taken is $A[i]$.

Additionally, when the flying squirrel passes through the horizontal position $i + 0.5$ ($0 \le i \le N-1$), the flying squirrel's height must be between $L[i]$ and $R[i]$ (inclusive). When the flying squirrel reaches the horizontal position N, its height must be exactly H.

Let $T[k]$ be the minimum total time taken to reach (N, H) satisfying all the above conditions with the number of times the flying squirrel jumped being exactly k. If such a way does not exist, define $T[k] = -1$.

Find $T[0], T[1], \cdots, T[H]$.

Implementation Details

You should implement the following function.

```
vector<long long> fly(int H, vector<int> A, vector<int> B, vector<int> L,
vector<int> R)
```

- H: The final height of the flying squirrel
- A, B, L, R: Integer arrays of size N
- B: An array representing the colors of the pillars. If $B[i] = 0$, the color of pillar i is red, and if $B[i] = 1$, the color of pillar i is blue.
- This function should return an array T of size $H + 1$.
- This function is called exactly once.

Constraints

- $1 \le N \le 200\,000$
- $0 \le H \le N$
- $0 \le A[i] \le 10^9$ $(0 \le i \le N-1)$
- $0 \le B[i] \le 1$ $(0 \le i \le N-1)$
- $0 \le L[i] \le R[i] \le N$ $(0 \le i \le N-1)$

Subtasks

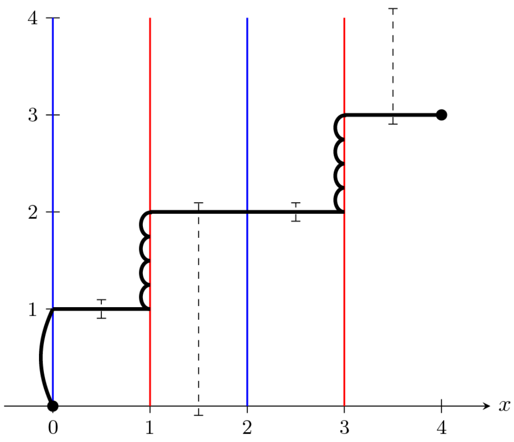
Number	Points	Constraints
1	3	$N \le 300$
2	4	$A[i] = B[i] = 0$ $(0 \le i \le N-1)$
3	25	$B[i] = 0$ $(0 \le i \le N-1)$
4	20	$N \le 65\,000$, $A[i] \le 5$ $(0 \le i \le N-1)$
5	29	$N \le 65\,000$
6	19	No additional constraints.

Examples

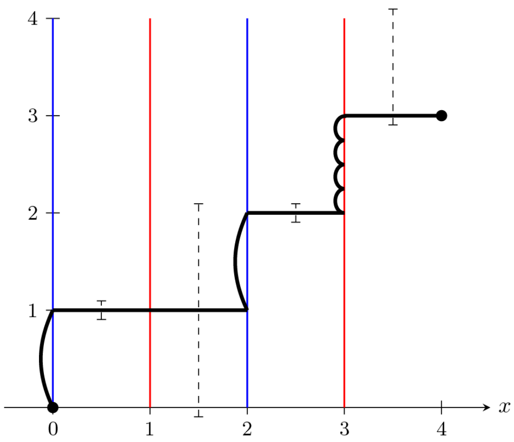
Example 1

Consider the following call.

```
fly(3, [8, 8, 2, 4],
       [1, 0, 1, 0],
       [1, 0, 2, 3],
       [1, 2, 2, 4])
```



If the flying squirrel jumps from pillar 0 and climbs pillars 1 and 3, the conditions are satisfied. In this case, the number of jumps is 1, and the time taken is 20 seconds.



If the flying squirrel jumps from pillars 0 and 2 and climbs pillar 3, the conditions are satisfied. In this case, the number of jumps is 2, and the time taken is 14 seconds.
No other ways are possible. Therefore, the function should return $[-1, 20, 14, -1]$.

Example 2

Consider the following call.

```
fly(1, [1000000000],
       [0],
       [1],
       [1])
```

The function should return $[1000000000, -1]$.

Example 3

Consider the following call.

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

The function should return $[7, -1, -1, -1]$.

Example 4

Consider the following call.

```
fly(7, [3, 3, 4, 1, 3, 2, 0, 1, 4, 3, 4, 0, 0, 1, 0, 4, 4, 5, 5, 0],
       [1, 1, 0, 0, 1, 1, 0, 1, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 1],
       [0, 0, 1, 1, 2, 1, 2, 2, 1, 1, 0, 1, 1, 3, 2, 2, 1, 6, 4, 4],
       [3, 2, 3, 3, 6, 2, 2, 4, 3, 4, 4, 5, 3, 6, 6, 5, 7, 8, 8, 9])
```

The function should return $[-1, 16, 11, 10, 9, 10, 12, 15]$.

Sample Grader

The input format for the sample grader is as follows:

- line 1: $N\ H$
- line 2: $A[0]\ A[1] \ldots A[N-1]$
- line 3: $B[0]\ B[1] \ldots B[N-1]$
- line 4: $L[0]\ L[1] \ldots L[N-1]$
- line 5: $R[0]\ R[1] \ldots R[N-1]$

The sample grader prints the answer in the following format:

- line 1: $T[0]\ T[1] \ldots T[H]$