나일강

당신은 나일강을 통해 N개 유물을 운반하려고 한다. 유물들은 0부터 $N-1$까지 번호가 붙어있다. 유물 i $(0 \leq i < N)$의 무게는 $W[i]$이다.

유물들을 운반하기 위해, 당신은 특별한 보트를 사용할 수 있다. 각 보트는 **최대 두 개**의 유물을 운반할 수 있다.

- 당신이 한 보트에 한 개의 유물을 운반하기로 결정하면, 유물의 무게와 상관 없이 항상 운반할 수 있다.
- 당신이 한 보트에 두 개의 유물을 운반하기로 결정하면, 보트의 균형이 고르게 잡히도록 해야 한다. 구체적으로, 당신이 한 보트에 유물 p와 q $(0 \leq p < q < N)$를 운반하기 위해서는, 두 유물의 무게 차이의 절댓값이 D 이하여야 한다. 다시 말해서, $|W[p] - W[q]| \leq D$.

유물을 운반하기 위해서, 당신은 같은 보트에 운반되는 유물의 수에 따라 결정되는 비용을 지불해야 한다. 유물 i $(0 \leq i < N)$를 운반하는 비용은 다음과 같다:

- 한 보트에 이 유물만을 운반하면, $A[i]$ 의 비용이 필요하다.
- 한 보트에 이 유물을 다른 유물과 같이 운반하면, $B[i]$의 비용이 필요하다.

후자의 경우에, 한 보트의 두 유물에 대해 모두 비용을 지불해야 한다는 점에 주목하자. 구체적으로, 당신이 한 보트에 유물 p와 q $(0 \leq p < q < N)$를 운반하기로 결정하면, 비용 $B[p] + B[q]$을 지불해야 한다.

보트에 하나의 유물만 운반하는 것은 두 개를 운반하는 것보다 항상 비싸다. 따라서 $0 \leq i < N$인 모든 i에 대해서, $B[i] < A[i]$.

불행히도, 강은 매우 예측 불가능하고, D의 값은 자주 바뀐다. 당신은 0부터 $Q-1$까지 번호 붙여진 Q개 질의에 답해야 한다. 질의들은 길이 Q의 배열 E로 표현된다. 질의 j $(0 \leq j < Q)$의 답은 D의 값이 $E[j]$일 때 N개 유물 모두를 운반하는 최소 비용이다.

Implementation Details

다음 함수를 구현해야 한다.

```
std::vector<long long> calculate_costs(
    std::vector<int> W, std::vector<int> A,
    std::vector<int> B, std::vector<int> E)
```

- W, A, B: 각각 유물들의 무게와 유물들을 운반하는 비용을 표현하는 길이 N의 정수 배열들.
- E: 각 질의에 대한 D의 값을 표현하는 길이 Q의 정수 배열.

- 이 함수는 유물들을 운반하는 최소 비용을 표현하는 Q개 정수들의 배열 R을 반환해야 한다. 여기서, $0 \leq j < Q$인 각 j에 대해서, $R[j]$는 D의 값이 $E[j]$일 때 그 비용이다.
- 이 함수는 각 테스트 케이스에 대해서 정확히 한번 호출된다.

Constraints

- $1 \leq N \leq 100\,000$
- $1 \leq Q \leq 100\,000$
- $0 \leq i < N$인 각 i에 대해, $1 \leq W[i] \leq 10^9$
- $0 \leq i < N$인 각 i에 대해, $1 \leq B[i] < A[i] \leq 10^9$
- $0 \leq j < Q$인 각 j에 대해, $1 \leq E[j] \leq 10^9$

Subtasks

Subtask	Score	Additional Constraints
1	6	$Q \leq 5$; $N \leq 2000$; $0 \leq i < N$인 각 i에 대해, $W[i] = 1$
2	13	$Q \leq 5$; $0 \leq i < N$인 각 i에 대해, $W[i] = i + 1$
3	17	$Q \leq 5$; $0 \leq i < N$인 각 i에 대해, $A[i] = 2$ 와 $B[i] = 1$
4	11	$Q \leq 5$; $N \leq 2000$
5	20	$Q \leq 5$
6	15	$0 \leq i < N$인 각 i에 대해, $A[i] = 2$ 와 $B[i] = 1$
7	18	추가적인 제약 조건이 없다.

Example

다음 호출을 생각해보자.

```
calculate_costs([15, 12, 2, 10, 21],
                [5, 4, 5, 6, 3],
                [1, 2, 2, 3, 2],
                [5, 9, 1])
```

이 예제에서, $N = 5$개 유물과 $Q = 3$개 질의가 있다.

첫번째 질의에서, $D = 5$. 당신은 한 보트에 유물 0과 3을 운반할 수 있고 ($|15 - 10| \leq 5$ 때문), 나머지 유물들은 한 보트에 한 개씩 운반한다. 이것이 모든 유물을 운반하는 최소 비용을 유도하고 이 최소 비용은 $1 + 4 + 5 + 3 + 3 = 16$이다.

두번째 질의에서, $D = 9$. 당신은 한 보트에 유물 0과 1을 운반하고($|15 - 12| \leq 9$ 때문) 한 보트에 유물 2와 3을 운반할 수 있다 ($|2 - 10| \leq 9$ 때문). 나머지 유물은 한 보트에 한 개씩 운반할 수 있다. 이것이 모든 유물을 운

반하는 최소 비용을 유도하고 이 최소 비용은 $1 + 2 + 2 + 3 + 3 = 11$이다.

마지막 질의에서, $D = 1$. 당신은 한 보트에 유물을 한 개씩만 운반한다. 이것이 모든 유물을 운반하는 최소 비용을 유도하고 이 최소 비용은 $5 + 4 + 5 + 6 + 3 = 23$이다.

그러므로, 이 함수는 $[16, 11, 23]$을 반환해야 한다.

Sample Grader

입력 형식:

```
N
W[0] A[0] B[0]
W[1] A[1] B[1]
...
W[N-1] A[N-1] B[N-1]
Q
E[0]
E[1]
...
E[Q-1]
```

출력 형식:

```
R[0]
R[1]
...
R[S-1]
```

여기서, S는 `calculate_costs`에 의해 반환된 배열 R의 길이다.