

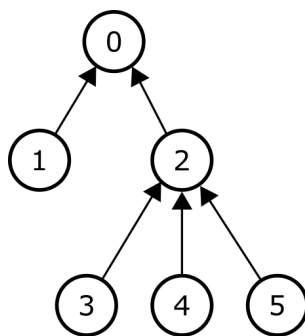
Tree

Дано корневое **дерево** на N вершинах, пронумерованных от 0 до $N - 1$. **Корнем** является вершина 0. У каждой вершины, кроме корня, есть единственный **родитель**. Для всех i , таких что $1 \leq i < N$, родителем i -й вершины является вершина $P[i]$, такая что $P[i] < i$. Будем полагать, что $P[0] = -1$.

Для каждого i ($0 \leq i < N$), **поддеревом** i -й вершины назовём набор из следующих вершин:

- i , и
- вершины, чей родитель — вершина i , и
- вершины, чей родитель родителя — вершина i , и
- вершины, чей родитель родителя родителя — вершина i , и
- так далее

На картинке ниже приведён пример дерева на $N = 6$ вершинах. Стрелка соединяет вершину с её родителем, кроме корня, у которого родителя нет. Поддерево вершины 2 состоит из вершин 2, 3, 4 и 5. Поддерево вершины 0 состоит из всех 6 вершин дерева, а поддерево вершины 4 состоит только из самой вершины 4.



Каждая вершина имеет неотрицательный целый **вес**. Обозначим за $W[i]$ вес i -й вершины ($0 \leq i < N$).

Ваша задача — написать программу, которая ответит на Q запросов, каждый из которых задаётся парой положительных целых чисел (L, R) . Ответ на запрос определяется следующим образом.

Для каждой вершины введём целое число, называемое **коэффициентом**. Получим последовательность $C[0], \dots, C[N - 1]$, где $C[i]$ ($0 \leq i < N$) — коэффициент, присвоенный i -й вершине. Назовём данную последовательность **последовательностью коэффициентов**.

Заметим, что элементы последовательности коэффициентов могут быть отрицательными, равными 0 или положительными.

Для запроса (L, R) , последовательность коэффициентов считается **корректной**, если для каждой вершины i ($0 \leq i < N$), выполняется следующее условие: сумма коэффициентов по всем вершинам поддерева i -й вершины не менее L и не более R .

Для последовательности коэффициентов $C[0], \dots, C[N - 1]$, **стоимость** i -й вершины будем считать равной $|C[i]| \cdot W[i]$, где $|C[i]|$ обозначает модуль числа $C[i]$. Наконец, **общей стоимостью** будем считать сумму стоимостей по всем вершинам. Ваша задача — для каждого запроса посчитать **минимальную общую стоимость**, которая может быть получена какой-либо корректной последовательностью коэффициентов.

Можно показать, что для любого запроса существует по крайней мере одна допустимая последовательность коэффициентов.

Implementation Details

Вы должны реализовать следующие две функции:

```
void init(std::vector<int> P, std::vector<int> W)
```

- P, W : массивы целых чисел длины N , задающие родителей и веса вершин.
- Данная функция будет вызвана ровно один раз, в начале взаимодействия грейдера и вашей программы.

```
long long query(int L, int R)
```

- L, R : целые числа, описывающие запрос.
- Данная функция будет вызвана Q раз после выполнения функции `init`.
- Функция должна вернуть ответ на данный запрос.

Constraints

- $1 \leq N \leq 200\,000$
- $1 \leq Q \leq 100\,000$
- $P[0] = -1$
- $0 \leq P[i] < i$ для всех i , таких что $1 \leq i < N$
- $0 \leq W[i] \leq 1\,000\,000$ для всех i , таких что $0 \leq i < N$
- $1 \leq L \leq R \leq 1\,000\,000$ для каждого запроса

Subtasks

| Подзадача | Баллы | Дополнительные ограничения |
|-----------|-------|---|
| 1 | 10 | $Q \leq 10$; $W[P[i]] \leq W[i]$ для всех i , таких что $1 \leq i < N$ |
| 2 | 13 | $Q \leq 10$; $N \leq 2\,000$ |
| 3 | 18 | $Q \leq 10$; $N \leq 60\,000$ |
| 4 | 7 | $W[i] = 1$ для всех i , таких что $0 \leq i < N$ |
| 5 | 11 | $W[i] \leq 1$ для всех i , таких что $0 \leq i < N$ |
| 6 | 22 | $L = 1$ |
| 7 | 19 | Без дополнительных ограничений. |

Examples

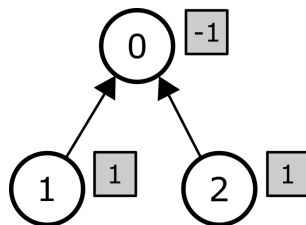
Рассмотрим следующие вызовы функций.

```
init([-1, 0, 0], [1, 1, 1])
```

Дерево состоит из 3 вершин: корень и 2 его ребёнка. Веса всех вершин равны 1.

```
query(1, 1)
```

В данном запросе $L = R = 1$, что означает, что сумма коэффициентов в поддереве каждой вершины обязана быть равна 1. Рассмотрим последовательность коэффициентов $[-1, 1, 1]$. Дерево и соответствующие коэффициенты (в тёмных прямоугольниках) изображены ниже.



Для каждой вершины i ($0 \leq i < 3$), сумма коэффициентов в поддереве i -й вершины равна 1. Поэтому последовательность коэффициентов является корректной. Общая стоимость считается следующим образом:

| Вершина | Вес | Коэффициент | Стоимость |
|---------|-----|-------------|--------------------|
| 0 | 1 | -1 | $ -1 \cdot 1 = 1$ |
| 1 | 1 | 1 | $ 1 \cdot 1 = 1$ |
| 2 | 1 | 1 | $ 1 \cdot 1 = 1$ |

Таким образом, общая стоимость равна 3. Это единственная корректная последовательность коэффициентов, поэтому функция должна вернуть 3.

```
query(1, 2)
```

Минимальная общая стоимость для этого запроса равна 2, и достигается на последовательности коэффициентов $[0, 1, 1]$.

Sample Grader

Input format:

```
N
P[1] P[2] ... P[N-1]
W[0] W[1] ... W[N-2] W[N-1]
Q
L[0] R[0]
L[1] R[1]
...
L[Q-1] R[Q-1]
```

где $L[j]$ и $R[j]$ ($0 \leq j < Q$) это аргументы для j -го вызова функции `query`. Заметим, что вторая строка содержит **только** $N - 1$ **целых чисел**, поскольку пример грейдера не считывает значение $P[0]$.

Output format:

```
A[0]
A[1]
...
A[Q-1]
```

где $A[j]$ (для $0 \leq j < Q$) — значение, которое возвращает j -й вызов функции `query`.