Träd

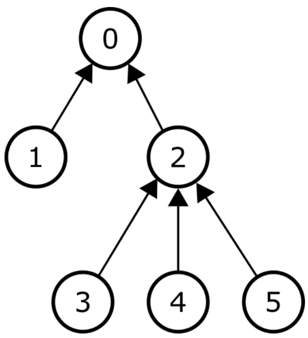
Betrakta ett **träd** som består av N **noder**, numrerade från 0 till $N-1$.

Nod 0 kallas för **roten**. Varje nod, förutom roten, har en enda **förälder**. För varje i, sådan att $1 \leq i < N$, gäller att föräldern till nod i är nod $P[i]$, där $P[i] < i$. Vi antar också att $P[0] = -1$.

För varje nod i ($0 \leq i < N$), så är **subträdet** för i mängden av följande noder:

- i, och
- alla noder vars förälder är i, och
- alla noder vars förälders förälder är i, och
- alla noder vars förälders förälders förälder är i,
- osv.

Bilden nedan visar ett exempelträd som består av $N = 6$ noder. Varje pil förbinder en nod med dess förälder, förutom roten som inte har någon förälder. Subträdet för nod 2 innehåller $2, 3, 4$ och 5 . Subträdet för nod 0 innehåller alla 6 noder i trädet och subträdet för nod 4 innehåller endast nod 4 .



Varje nod tilldelas en **vikt** som är ett icke-negativt heltal. Vi betecknar vikten av nod i ($0 \leq i < N$) som $W[i]$.

Din uppgift är att skriva ett program som kommer att svara på Q frågor, där varje fråga består av ett par av heltal (L, R). Svaret på frågan bör beräknas enligt följande.

Låt oss tilldela ett heltal som kallas en **koefficient**, till varje nod i trädet. En sådan tilldelning beskrivs av en sekvens $C[0], \ldots, C[N-1]$, där $C[i]$ ($0 \leq i < N$) är koefficienten som är tilldelad nod i . Låt oss kalla denna sekvens för en **koefficientsekvens**. Observera att elementen i koefficientsekvensen kan vara negativa, 0 eller positiva.

För en fråga (L, R), är en koefficientsekvens **giltig** om, för varje nod i ($0 \leq i < N$), gäller följande villkor: summan av koefficienterna för noderna i subträdet till nod i är inte mindre än L och inte större än R.

För en given koefficientsekvens $C[0], \ldots, C[N-1]$, så är **kostnaden** för en nod i lika med $|C[i]| \cdot W[i]$, där $|C[i]|$ anger det absoluta värdet av $C[i]$. Slutligen är **totalkostnaden** summan av kostnaderna för alla noder. Din uppgift är att för varje fråga beräkna den **minsta totalkostnaden** som kan uppnås med någon giltig koefficientsekvens.

Det kan bevisas att för varje fråga finns åtminstone en giltig koefficientsekvens.

Implementationsdetaljer

Harry rekommenderar starkt att du implementerar följande två funktioner:

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

- P, W: arrayer med heltal av längd N som anger föräldrarna och vikterna för varje nod.
- Denna procedur kallas exakt en gång i början av interaktionen mellan gradern och ditt program i varje testfall.

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

- L, R: heltal som beskriver en fråga.
- Denna procedur kallas Q gånger efter anropet av `init` i varje testfall.
- Denna procedur bör returnera svaret på den givna frågan.

Begränsningar

- $1 \leq N \leq 200\,000$
- $1 \leq Q \leq 100\,000$
- $P[0] = -1$
- $0 \leq P[i] < i$ för alla i sådan att $1 \leq i < N$
- $0 \leq W[i] \leq 1\,000\,000$ för alla i sådan att $0 \leq i < N$
- $1 \leq L \leq R \leq 1\,000\,000$ för varje fråga

Subtasks

Grupp	Poäng	Ytterligare begränsningar
1	10	$Q \leq 10$; $W[P[i]] \leq W[i]$ för alla i sådan att $1 \leq i < N$
2	13	$Q \leq 10$; $N \leq 2\,000$
3	18	$Q \leq 10$; $N \leq 60\,000$
4	7	$W[i] = 1$ för alla i sådan att $0 \leq i < N$
5	11	$W[i] \leq 1$ för alla i sådan att $0 \leq i < N$
6	22	$L = 1$
7	19	Inga ytterligare begränsningar.

Examplen

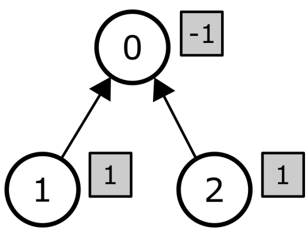
Pondera på följande anrop:

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

Trädet består av 3 noder: roten och rotens 2 barn. Alla noder har vikten 1.

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

I den här frågan gäller $L = R = 1$, vilket innebär att summan av koefficienterna i varje subträd måste vara lika med 1. Låt oss använda koefficientsekvensen $[-1, 1, 1]$. Trädet och dess korresponderande koefficienter (i skuggade rektanglar) är illusterade nedan.



För varje nod i ($0 \leq i < 3$), är summan av alla koefficienter av alla noder i subträdet för i lika med 1. Därför är denna koefficientsekvensen giltig. Den totala kostnaden är beräknad på följande vis:

Nod	Vikt	Koefficient	Kostnad
0	1	-1	$\mid -1 \mid \cdot 1 = 1$
1	1	1	$\mid 1 \mid \cdot 1 = 1$
2	1	1	$\mid 1 \mid \cdot 1 = 1$

Den totala kostnaden för det här fallet är 3. Det här är även den enda giltiga koefficientsekvensen, därför borde detta anrop returnera 3.

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

Den minsta totala kostnaden för denna fråga är 2, vilket kan uppnås genom att använda koefficientsekvensen $[0, 1, 1]$.

Exempelgrader

Inputformat:

```
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]
```

där $L[j]$ och $R[j]$ (för $0 \leq j < Q$) är funktionsparametrarna i det j:te anropet till `query` . Observera att den andra raden i inmatningen innehåller **endast** $N - 1$ **heltal**, eftersom exempelgradern inte läser värdet på $P[0]$.

Outputformat:

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

där $A[j]$ (för $0 \leq j < Q$) är värdet som returneras av j:te anropet till `query`.