

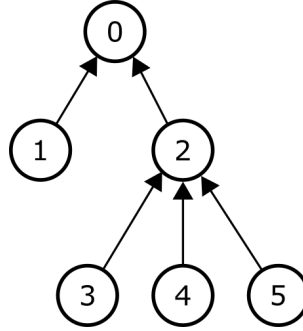
Ağaç

0'dan $N - 1$ 'e kadar numaralandırılmış N düğümden oluşan bir **ağaç** düşünün. Düğüm 0'a **kök** denir. Kök hariç her düğümün tek bir **ebeveyni** vardır. $1 \leq i < N$ olacak şekilde her i için, i düğümünün ebeveyni $P[i]$ düğümüdür, burada $P[i] < i$. Ayrıca $P[0] = -1$ olduğunu varsayıyoruz.

Herhangi bir düğüm i ($0 \leq i < N$) için, i nin **alt ağacı** aşağıdaki düğümlerin kümesidir:

- i ve
- ebeveyni i olan herhangi bir düğüm ve
- ebeveyninin ebeveyni i olan herhangi bir düğüm ve
- ebeveyninin ebeveyninin ebeveyni i olan herhangi bir düğüm ve
- vs.

Aşağıdaki şekil $N = 6$ düğümden oluşan örnek bir ağacı göstermektedir. Her kenar, ebeveyni olmayan kök hariç olmak üzere, bir düğümü ebeveynine bağlar. 2 düğümünün alt ağacı 2, 3, 4 ve 5 düğümlerini içerir. 0 düğümünün alt ağacı, ağacın tüm 6 düğümünü içerir ve 4 düğümünün alt ağacı yalnızca 4 düğümünü içerir.



Her düğüme negatif olmayan bir **ağırlık** atanır. i ($0 \leq i < N$) düğümünün ağırlığını $W[i]$ ile gösteriyoruz.

Göreviniz, her biri bir çift tam sayı (L, R) ile belirtilen, Q sorguya cevap verecek bir program yazmaktır. Sorgunun cevabı aşağıdaki gibi hesaplanmalıdır.

Ağacın her bir düğümüne **katsayı** adı verilen bir tam sayı atamayı düşünün. Böyle bir atama $C[0], \dots, C[N - 1]$ serisi ile tanımlanır, burada $C[i]$ ($0 \leq i < N$) i düğümüne atanan katsayıdır. Bu seriye **katsayı serisi** diyelim. Katsayı serisinin elemanlarının negatif, 0 veya pozitif olabileceğini unutmayın.

(L, R) sorgusu için, bir katsayı serisine **geçerli** denir, eğer her i düğümü ($0 \leq i < N$) için, aşağıdaki koşul geçerli ise: i düğümünün alt ağacındaki düğümlerin katsayılarının toplamı L 'den küçük ve R 'den büyük değildir.

Verilen bir katsayı serisi $C[0], \dots, C[N - 1]$ için, bir i düğümünün **maliyeti** $|C[i]| \cdot W[i]$ dir, burada $|C[i]|$, $C[i]$ nin mutlak değerini ifade eder. Son olarak, **toplam maliyet** tüm düğümlerin maliyetlerinin toplamıdır. Her sorgu için göreviniz, geçerli bir katsayı serisiyle elde edilebilecek **minimum toplam maliyeti** hesaplamaktır.

Gerçekleştirim Detayları

Aşağıdaki iki prosedürü kodlamalısınız:

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

- P, W : N uzunluğunda ebeveynleri ve ağırlıkları belirten tam sayı dizileri.
- Bu prosedür her test durumunda, grader ile programınız arasındaki etkileşimin başlangıcında tam olarak bir kez çağrılır.

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

- L, R : Bir sorguyu temsil eden tam sayılar.
- Bu prosedür her test durumunda `init` çağrısından sonra Q kez çağrılır.
- Bu prosedür verilen sorguya cevap dönmelidir.

Kısıtlar

- $1 \leq N \leq 200\,000$
- $1 \leq Q \leq 100\,000$
- $P[0] = -1$
- Her i için $0 \leq P[i] < i$ öyle ki $1 \leq i < N$
- Her i için $0 \leq W[i] \leq 1\,000\,000$ öyle ki $0 \leq i < N$
- Her sorguda $1 \leq L \leq R \leq 1\,000\,000$

Altgörevler

Altgörev	Puan	Ek kısıtlar
1	10	Her i için $Q \leq 10$; $W[P[i]] \leq W[i]$ öyle ki $1 \leq i < N$
2	13	$Q \leq 10$; $N \leq 2\,000$
3	18	$Q \leq 10$; $N \leq 60\,000$
4	7	Her i için $W[i] = 1$ öyle ki $0 \leq i < N$
5	11	Her i için $W[i] \leq 1$ öyle ki $0 \leq i < N$
6	22	$L = 1$
7	19	Ek kısıt yoktur.

Örnekler

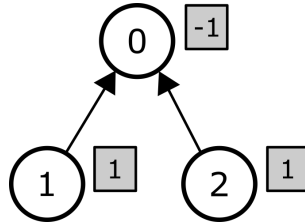
Aşağıdaki çağrıyı göz önüne alın:

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

Ağaç kök ve kökün 2 çocuğu olmak üzere 3 düğümden oluşur. Tüm düğümlerin ağırlığı 1 dir.

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

Bu sorguda $L = R = 1$, bu da her alt ağaçtaki katsayıların toplamının 1 e eşit olması gerektiği anlamına gelir. $[-1,1,1]$ katsayı serisini ele alalım. Ağaç ve karşılık gelen katsayılar (gölgeli dikdörtgenler) aşağıda gösterilmiştir.



Her düğüm i ($0 \leq i < 3$) için, i nin alt ağacındaki tüm düğümlerin katsayılarının toplamı 1 e eşittir. Dolayısıyla bu katsayı serisi geçerlidir. Toplam maliyet aşağıdaki şekilde hesaplanır:

Düğüm	Ağırlık	Katsayı	Maliyet
0	1	-1	$ -1 \cdot 1 = 1$
1	1	1	$ 1 \cdot 1 = 1$
2	1	1	$ 1 \cdot 1 = 1$

Dolayısıyla toplam maliyet 3 dür. Bu geçerli tek katsayı serisidir. bu nedenle bu çağrı 3 değerini dönmelidir.

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

Bu sorgu için minimum toplam maliyet 2 dir, ve bu katsayı serisi $[0, 1, 1]$ olduğunda elde edilir.

Örnek Değerlendirici (Sample Grader)

Girdi formatı:

```
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]
```

burada $L[j]$ ve $R[j]$ ($0 \leq j < Q$ için) query çağrısının j -inci çağrısındaki girdi argümanlarıdır. Örnek değerlendirici $P[0]$ değerini okumadığından, girdinin ikinci satırının **sadece $N - 1$ tam sayı** içerdiğine dikkat edin,

Çıktı formatı:

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

burada $A[j]$ ($0 \leq j < Q$ için) query çağrısının j -inci çağrısında dönen değerdir.