

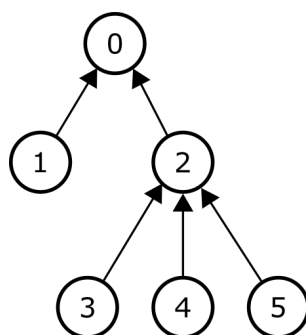
# Tree

Considerăm un **arbore** cu  $N$  **noduri**, numerotate de la 0 la  $N - 1$ . Nodul 0 este numit **rădăcină**. Fiecare nod, cu excepția rădăcinii, are un singur **părinte**. Pentru fiecare  $i$ ,  $1 \leq i < N$ , părintele nodului  $i$  este nodul  $P[i]$ , unde  $P[i] < i$ . De asemenea, considerăm  $P[0] = -1$ .

Pentru orice nod  $i$  ( $0 \leq i < N$ ), **subarborele** nodului  $i$  este mulțimea formată din următoarele noduri:

- $i$ , și
- orice nod al cărui părinte este  $i$ , și
- orice nod părintele părintelui căruia este  $i$ , și
- orice nod părintele părintelui părintelui căruia este  $i$ , și
- tot așa.

În imaginea care urmează este prezentat un exemplu de arbore cu  $N = 6$  noduri. Fiecare arc conectează un nod la părintele său, cu excepția rădăcinii, care nu are părinte. Subarborele nodului 2 este format din nodurile 2, 3, 4 și 5. Subarborele nodului 0 este format din toate cele 6 noduri ale arborelui și subarborele nodului 4 conține doar nodul 4.



Fiecărui nod îi este atașată o **greutate** nenegativă, întreagă. Vom nota greutatea nodului  $i$  ( $0 \leq i < N$ ) prin  $W[i]$ .

Sarcina ta este să scrii un program care va răspunde la  $Q$  interogări, fiecare fiind descrisă de o pereche de valori întregi, pozitive ( $L, R$ ). Răspunsul la interogare urmează să fie calculat după cum urmează:

Fiecărui nod îi considerăm asociat un întreg, numit **coeficient**. Această asociere este descrisă de o secvență  $C[0], \dots, C[N - 1]$ , unde  $C[i]$  ( $0 \leq i < N$ ) este coeficientul atribuit nodului  $i$ . Vom numi această secvență **secvența de coeficienți**. De notat, că elementele secvenței de coeficienți pot avea valori negative, 0, sau pozitive.

Pentru interogarea  $(L, R)$ , secvența de coeficienți este considerată **validă** dacă, pentru orice nod  $i$  ( $0 \leq i < N$ ), are loc următoarea condiție: suma coeficienților nodurilor din subarborele nodului  $i$  nu este mai mică decât  $L$  și nu depășește  $R$ .

Pentru o secvență de coeficienți dată  $C[0], \dots, C[N-1]$ , **costul** nodului  $i$  este  $|C[i]| \cdot W[i]$ , unde  $|C[i]|$  înseamnă valoarea absolută a  $C[i]$ . La final, **costul total** este suma costurilor pentru toate nodurile. Sarcina ta este să calculezi, pentru fiecare interogare, **costul total minimal** care poate fi obținut de o secvență de coeficienți validă.

Se poate demonstra că pentru orice interogare există cel puțin o secvență de coeficienți validă.

## Detalii de Implementare

Urmează să implementezi următoarele două funcții:

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

- $P, W$ : tablouri de lungime  $N$  conținând numere întregi specificând părinții și greutatea.
- Această funcție este apelată exact o singură dată la începutul interacțiunii între grader și programul tău în fiecare test.

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

- $L, R$ : numere întregi care descriu interogarea.
- Această funcție este apelată de  $Q$  ori după invocarea funcției `init` în fiecare test.
- Această funcție va returna răspunsul la o interogare dată.

## Restricții

- $1 \leq N \leq 200\,000$
- $1 \leq Q \leq 100\,000$
- $P[0] = -1$
- $0 \leq P[i] < i$  pentru fiecare  $i$ ,  $1 \leq i < N$
- $0 \leq W[i] \leq 1\,000\,000$  pentru fiecare  $i$ ,  $0 \leq i < N$
- $1 \leq L \leq R \leq 1\,000\,000$  în fiecare interogare

## Subtask-uri

Subtask	Punctaj	Restricții adiționale
1	10	$Q \leq 10$ ; $W[P[i]] \leq W[i]$ pentru fiecare $i$ , $1 \leq i < N$
2	13	$Q \leq 10$ ; $N \leq 2\,000$
3	18	$Q \leq 10$ ; $N \leq 60\,000$
4	7	$W[i] = 1$ pentru fiecare $i$ , $0 \leq i < N$
5	11	$W[i] \leq 1$ pentru fiecare $i$ , $0 \leq i < N$
6	22	$L = 1$
7	19	Fără restricții adiționale.

## Exemple

Considerăm următoarele apeluri:

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

Arborele conține 3 noduri, rădăcina și cei doi copii ai ei. Toate nodurile au greutatea 1.

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

În această interogare  $L = R = 1$ , ceea ce înseamnă că suma coeficienților în fiecare subarbore trebuie să fie egală cu 1. Considerăm secvența de coeficienți  $[-1, 1, 1]$ . Arborele și coeficienții corespunzători (în dreptunghiurile umbrite) sunt ilustrate mai jos.



Pentru fiecare nod  $i$  ( $0 \leq i < 3$ ), suma coeficienților tuturor nodurilor din subarborele \$i\$ este egală cu 1. Astfel, această secvență de coeficienți este validă. Costul total se calculează după cum urmează:

Nod	Greutate	Coeficient	Cost
0	1	-1	$ -1  \cdot 1 = 1$
1	1	1	$ 1  \cdot 1 = 1$
2	1	1	$ 1  \cdot 1 = 1$

Prin urmare, costul total este 3. Aceasta este unica secvență de coeficienți validă, prin urmare acest apel trebuie să returneze 3.

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

Costul total minimal pentru această interogare este 2, și se obține atunci când secvența de coeficienți este  $[0, 1, 1]$ .

## Grader local

Format intrare:

```
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]
```

unde  $L[j]$  și  $R[j]$  (pentru  $0 \leq j < Q$ ) sunt argumentele de intrare în a  $j$ -ul apel către interogare. De notat că cea de a doua linie din intrare conține **doar**  $N - 1$  **intregi**, iar graderul local nu va citi valoarea  $P[0]$ .

Format ieșire:

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

unde  $A[j]$  (pentru  $0 \leq j < Q$ ) este valoarea returnată de apelul al  $j$ -lea către interogare.