

# Tree

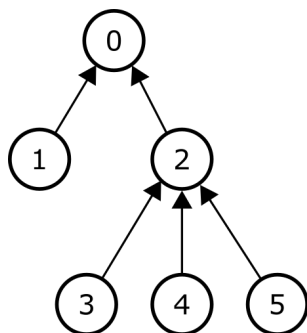
Konsideroni një **pemë** me  $N$  kulme **kulme**, nga 0 deri në Kulmi  $N - 1$ . Vertex 0 quhet rrënja **rrënja** e pemës. Çdo kulm, përveç rrënjës ka një prind **prind** të vetëm. Për çdo  $i$ ,  $1 \leq i < N$ , , prindi i kulmit  $i$  është kulmi  $P[i]$ , ku  $P[i] < i$ . Gjithashtu supozojmë se  $P[0] = -1$ .

Për çdo kulm  $i$  ( $0 \leq i < N$ ), **nënpema** e  $i$  është bashkësia e kulmeve si më poshtë:

- $i$ , dhe
- çdo kulm me prind  $i$ , dhe
- çdo kulm ku prindi i prindit është  $i$ , dhe
- çdo kulm ku prindi i prindit të prindit është  $i$ , dhe
- etj.

Foto më poshtë tregon një shembull të pemës që përbëhet nga  $N = 6$  kulme. Çdo shigjetë lidh kulmin me prindin e tij, përveç rrënjës e cila nuk ka prind.

Nënpema e kulmit 2 përmban kulmet 2, 3, 4 dhe 5. Nënpema e kulmit 0 përmban të 6 kulmet e pemës dhe nënpema e kulmit 4 përmban vetëm kulmin 4.



Çdo kulm është shënuar me **peshë** jonegative. Ne përcaktojmë peshën e kulmit  $i$  ( $0 \leq i < N$ ) si  $W[i]$ .

Detyra juaj është të shkruani një program i cili do të jape përgjigje për  $Q$  queries, secila e përcaktuar nga një çift numrash të plotë  $(L, R)$ . Përgjigjet e query duhet të llogariten si më poshtë.

Konsideroni të caktoni një numër të plotë për secilën kënd të pemës , të quajtur **koeficient**,. Një detyrë e tillë përshkruhet nga një sekuencë  $C[0], \dots, C[N - 1]$ , ku  $C[i]$  ( $0 \leq i < N$ ) është koeficienti i caktuar për këndin  $i$ . Le ta quajmë këtë sekuencë një **sekuencë koeficienti**. Elementët e sekuencës së koeficientit mund të jenë negativ, 0, ose pozitivë.

Për query  $(L, R)$ , si sekuencë koeficienti quhet i v **vlefshëm** në qoftë se, për çdo kulm  $i$  ( $0 \leq i < N$ ), vlen kushti i mëposhtëm : shuma e koeficientëve të kulmeve në nënpemën e kulmit  $i$  është jo më e vogël se  $L$  dhe jo më e madhe se  $R$ .

Për sekuencën e koeficientit të dhënë  $C[0], \dots, C[N - 1]$ , **kostoja** e kulmit  $i$  është  $|C[i]| \cdot W[i]$ , ku  $|C[i]|$  tregon vlerën absolute të  $C[i]$ . **Kostoja totale** është sa shuma e kostove për të gjithë këndet. Detyra juaj është të llogarisni për çdo query **koston totale minimale** që mund të arrihet me një sekuencë koeficienti të vlefshme.

## Implementation Details

Duhet të zbatohen dy procedurat më poshtë:

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

- $P, W$ : arrays me numra të plotë të gjatësis  $N$  specifikon prindin dhe peshën.
- Procedura thërret një herë në fillim.

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

- $L, R$ : numër i plotë që përshkruan një query.
- Procedura thërret  $Q$  herë pas thirrjes së `init` në çdo rast tetsi.
- Procedura do japi vlerën e query së dhënë.

## Constraints

- $1 \leq N \leq 200\,000$
- $1 \leq Q \leq 100\,000$
- $P[0] = -1$
- $0 \leq P[i] < i$  for each  $i$  such that  $1 \leq i < N$
- $0 \leq W[i] \leq 1\,000\,000$  for each  $i$  such that  $0 \leq i < N$
- $1 \leq L \leq R \leq 1\,000\,000$  in each query

## Subtasks

Subtask	Score	Additional Constraints
1	10	$Q \leq 10; W[P[i]] \leq W[i]$ for each $i$ such that $1 \leq i < N$
2	13	$Q \leq 10; N \leq 2\,000$
3	18	$Q \leq 10; N \leq 60\,000$
4	7	$W[i] = 1$ for each $i$ such that $0 \leq i < N$
5	11	$W[i] \leq 1$ for each $i$ such that $0 \leq i < N$
6	22	$L = 1$
7	19	No additional constraints.

## Examples

Consider the following calls:

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

Pema ka 3 kulme, ka rrenjët dhe 2 femij. Kulmet kan gjerësi 1.

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

Kjo query  $L = R = 1$ , që do të thotë se shuma e koeficientit në çdo nënpemë duhet të jetë e barabartë me 1. Konsideroni koeficientin me sekuencë  $[-1, 1, 1]$ . Pemët dhe koeficienti përkatës afishohen më poshtë.



Për çdo kulm  $i$  ( $0 \leq i < 3$ ), shuma e koeficientit të kulmeve në nënpemë të  $i$  është e barabartë me 1. Prandaj, sekuenca e koeficientit është valide t. The total cost is computed as follows:

Vertex	Weight	Coefficient	Cost
0	1	-1	$ -1  \cdot 1 = 1$
1	1	1	$ 1  \cdot 1 = 1$
2	1	1	$ 1  \cdot 1 = 1$

Therefore the total cost is 3. This is the only valid coefficient sequence, therefore this call should return 3.

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

The minimum total cost for this query is 2, and is attained when the coefficient sequence is  $[0, 1, 1]$ .

## 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]
```

where  $L[j]$  and  $R[j]$  (for  $0 \leq j < Q$ ) are the input arguments in the  $j$ -th call to query. Note that the second line of the input contains **only**  $N - 1$  **integers**, as the sample grader does not read the value of  $P[0]$ .

Output format:

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

where  $A[j]$  (for  $0 \leq j < Q$ ) is the value returned by the  $j$ -th call to query.