

Yana daraxtlar (tree)

Yangi yil arafasida Komiljon tog'ga chiqib ajoyib yangi yil daraxtini topdi. Daraxtning go'zalligini ko'rgan Xurshid quyidagi masalani o'ylab topdi.

Sizga n ta tugunli **ildizli** daraxt berilgan. Daraxtning tugunlari 0 dan $n - 1$ gacha raqamlangan, daraxt ildizi esa r -raqamli tugun. Barcha $i \neq r$ uchun i -tugunning otasi $p[i]$ -raqamli tugun.

Ixtiyoriy a va b tugunlar uchun, $\text{lca}(a, b)$ deb, shu tugunlarning eng kichkina umumiy otasiga aytiladi. Xususan, ixtiyoriy x uchun $\text{lca}(x, x) = x$.

$f(a, b)$ ni quyidagicha ifodalaylik:

$$f(a, b) = a \cdot n^2 + \text{lca}(a, b) \cdot n + b$$

Shuningdek, barcha (a, b) juftliklar uchun $f(a, b)$ ning qiymatini avvaliga bo'sh bo'lgan s massivga qo'shib chiqaylik va kamaymaslik tartibida saralab olaylik. Bunda $a = b$ holatlar ham hisobga olinadi, ya'ni s massivda aynan n^2 ta element bor.

Sizga q ta so'rovda k sonlari beriladi. Har bir so'rov uchun: saralangan s massivining k -elementini toping (indekslar 0 dan boshlanadi).

Implementation details

Siz quyidagi protsedurani dasturlashingiz kerak:

```
vector<long long> answer(int n, int r, vector<int> p, int q, vector<long long> k);
```

- n : daraxtdagi tugunlar soni.
- r : daraxtning ildizi.
- p : uzunligi n bo'lgan massiv – daraxt haqida ma'lumot. $p[r] = -1$.
- q : jami so'rovlar soni.
- k : uzunligi q bo'lgan massiv – so'rovlar haqida ma'lumot.
- Bu protsedura uzunligi q bo'lgan massiv qaytarishi kerak, bunda j -element j -so'rov uchun javob.
- Bu protsedura aynan bir marta chaqiriladi.

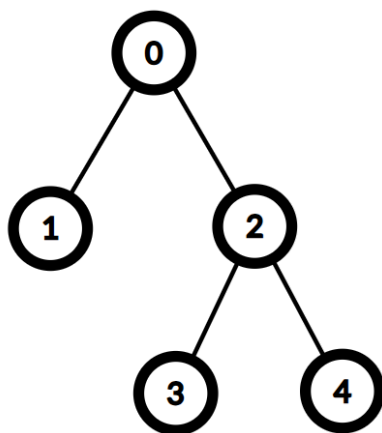
Example

Example 1

Quyidagi chaqiruvni ko'raylik:

```
answer(5, 0, [-1, 0, 0, 2, 2], 3, [11, 17, 3])
```

Daraxt quyidagi ko'rinishda:



Saralagandan so'ng, s quyidagi ko'rinishga keladi:
 $s = [0, 1, 2, 3, 4, 25, 27, 28, 29, 31, 50, 51, 62, 63, 64, 75, 76, 87, 89, 93, 100, 101, 112, 113, 124]$.

$q = 3$ ta so'rovlar quyidagicha:

- $k[0] = 11, s[11] = 51$. Bu javob quyidagi ko'rinishda olinadi: $f(2, 1) = 2 \cdot 5^2 + 0 \cdot 5 + 1 = 51$
- $k[1] = 17, s[17] = 87$. Bu javob quyidagi ko'rinishda olinadi: $f(3, 2) = 3 \cdot 5^2 + 2 \cdot 5 + 2 = 87$
- $k[2] = 3, s[3] = 3$. Bu javob quyidagi ko'rinishda olinadi: $f(0, 4) = 0 \cdot 5^2 + 0 \cdot 5 + 4 = 4$

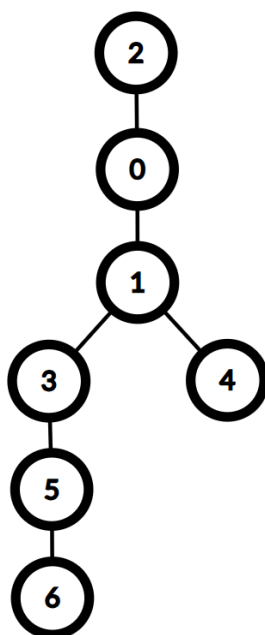
Protsedura javob sifatida $[51, 87, 4]$ massivini qaytarishi kerak.

Example 2

Quyidagi chaqiruvni ko'raylik:

```
answer(7, 2, [2, 0, -1, 1, 1, 3, 5], 7, [30, 2, 46, 38, 15, 31, 30])
```

Daraxt quyidagi ko'rinishda:



Protsedura javob sifatida [206, 3, 318, 261, 113, 208, 206] massivini qaytarishi kerak.

Constraints

- $2 \leq n \leq 10^6$
- $0 \leq r \leq n - 1$
- $0 \leq p[i] \leq n - 1$ (barcha $0 \leq i < n$ va $i \neq r$ uchun)
- $p[r] = -1$
- $1 \leq q \leq 10^6$
- $0 \leq k[j] \leq n^2 - 1$ (barcha $0 \leq j \leq q - 1$ uchun)

Subtasks

d deb daraxtning chuqurligini, ya'ni ildizdan istalgan tugungacha maksimal masofani aytaylik.

Qaysidir v tugunning darajasi bu v ga bog'langan boshqa tugunlar soni bo'lsin. e deb barcha tugunlar orasidagi maksimal darajaga aytiladi.

1. (7 ball) $n, q \leq 300$
2. (13 ball) $n, q \leq 2000$
3. (13 ball) $n, q \leq 10^5, d \leq 20$
4. (17 ball) $d \leq 20$
5. (15 ball) $e \leq 2$
6. (13 ball) $n, q \leq 10^5$
7. (22 ball) Qo'shimcha cheklavlarsiz.

Sample Grader

Namunaviy grader ma'lumotlarni quyidagi tartibda o'qiydi:

- qator 1: n r
- qator 2: $p[0] \ p[1] \ \dots \ p[n - 1]$
- qator 3: q
- qator $3 + j$ ($0 \leq j < q$): $k[j]$

Aytaylik, `answer()` protsedurasi javob sifatida t massivni qaytarsin. Namunaviy grader javoblarni quyidagi tartibda chiqaradi:

- qator $1 + j$ ($0 \leq j < q$): $t[j]$