

## 集宝 (gems)

### 【题目描述】

第一千届收集宝石大赛开始了!

观众们都厌倦了发生在线性序列上的集宝问题,因此本次大赛相比历届有着明显的创新:参赛选手现在需要在一棵  $n$  个结点的无根树上收集宝石。

这棵树上一共有  $m$  颗宝石,其中第  $i$  ( $1 \leq i \leq m$ ) 颗宝石拥有相应的二元组参数  $(a_i, d_i)$ ,表示当一名选手当前到达结点  $a_i$  的简单路径所含边数不超过  $d_i$  时,他就可以选择立刻收集这颗宝石(当然,也可以选择收集)。

相应地,本次大赛的选手也热情高涨,总共有  $q$  名选手报名参赛。对于每名选手,他将会被分配一个出发结点  $x$  与一个收集宝石的区间  $[l, r]$ ,他需要完成以下任务:从结点  $x$  出发,不断选择当前结点的一条邻边并移动过去,依次收集第  $l$  至第  $r$  颗宝石,即按照  $l, l+1, \dots, r$  的顺序收集所有  $r-l+1$  颗宝石。

因为每名选手在参赛过程中都遇到了或多或少的路线规划难题,为了给出合理的评分,主办方找到了你。请你对每一名选手,计算出该选手完成收集任务所需移动的最小总边数。

### 【实现细节】

选手不需要,也不应该实现 `main` 函数。

选手需要确保提交的程序包含头文件 `gems.h`,即在程序开头加入以下代码:

```
1 #include "gems.h"
```

选手需要在提交的程序源文件 `gems.cpp` 中实现以下两个函数:

```
1 void gems(int c, int n, int m, std::vector<int> u,
           std::vector<int> v, std::vector<int> a, std::vector<int>
           d);
```

- $c, n, m$  分别表示测试点编号、树的结点数与宝石的数量。 $c = 0$  表示该测试点为样例。
- 对于  $0 \leq i < n - 1$ ,  $u_i, v_i$  表示树的一条边。
- 对于  $0 \leq i < m$ ,  $a_i, d_i$  表示第  $i + 1$  颗宝石的两个参数。
- 对于每个测试点,该函数会被交互库调用恰好一次,且在所有 `query` 函数调用之前。

```
1 long long query(int x, int l, int r);
```

- $x, l, r$  分别表示一名选手的出发结点与收集宝石的区间。

- 该函数需要返回该选手从结点  $x$  出发, 依次收集第  $l$  至第  $r$  颗宝石所需移动的最小总边数。
- 对于每个测试点, 该函数会被交互库调用恰好  $q$  次。

### 【测试程序方式】

选手可以在本题目录下使用如下命令编译得到可执行程序:

```
1 g++ grader.cpp gems.cpp -o gems -O2 -std=c++14 -static
```

对于编译得到的可执行程序:

- 可执行文件将从标准输入读入以下格式的数据:
  - 输入的第一行包含三个非负整数  $c, n, m$ 。
  - 输入的第  $1 + i$  ( $1 \leq i \leq n - 1$ ) 行包含两个正整数  $u_i, v_i$ 。
  - 输入的第  $n + 1$  行包含  $m$  个正整数  $a_1, a_2, \dots, a_m$ 。
  - 输入的第  $n + 2$  行包含  $m$  个非负整数  $d_1, d_2, \dots, d_m$ 。
  - 输入的第  $n + 3$  行包含一个正整数  $q$ 。
  - 输入的第  $n + 3 + i$  ( $1 \leq i \leq q$ ) 行包含三个正整数  $x, l, r$ 。
- 可执行文件将输出以下格式的数据至标准输出:
  - 输出共  $q$  行, 每行一个非负整数, 表示 `query` 函数的返回值。

### 【样例 1 输入】

```
1 0 5 4
2 1 2
3 1 3
4 2 4
5 2 5
6 4 1 5 3
7 1 0 1 2
8 2
9 2 2 4
10 4 1 2
```

### 【样例 1 输出】

```
1 2
2 2
```

**【样例 1 解释】**

对于第一名选手，该选手需要从结点 2 出发，收集第 2 ~ 4 颗宝石。一种可能的收集路径为  $2 \rightarrow 1 \rightarrow 2 \rightarrow 2$ ，共经过 2 条边。

对于第二名选手，该选手需要从结点 4 出发，收集第 1 ~ 2 颗宝石。一种可能的收集路径为  $4 \rightarrow 4 \rightarrow 1$ ，共经过 2 条边。

**【数据范围】**

对于所有测试数据，均有：

- $2 \leq n \leq 3 \times 10^5$ ,  $1 \leq m \leq 3 \times 10^5$ ;
- 对于所有  $1 \leq i \leq n - 1$ ，均有  $1 \leq u_i, v_i \leq n$ ，且所有的边构成一棵树；
- 对于所有  $1 \leq i \leq m$ ，均有  $1 \leq a_i \leq n$  且  $0 \leq d_i \leq n$ ;
- $1 \leq q \leq 5 \times 10^5$ ;
- $1 \leq x \leq n$ ,  $1 \leq l \leq r \leq m$ 。

测试点编号	$n, m \leq$	$q \leq$	特殊性质
1 ~ 3	$10^2$	$10^2$	无
4 ~ 7	$10^3$	$10^3$	
8 ~ 10		$3 \times 10^5$	
11, 12	$3 \times 10^5$	$5 \times 10^5$	A
13, 14			B
15 ~ 17			C
18 ~ 21	$10^5$	$3 \times 10^5$	无
22 ~ 25	$3 \times 10^5$	$5 \times 10^5$	

特殊性质 A：树的形态是一条链，即不存在度数超过 2 的结点。

特殊性质 B：对于所有  $1 \leq i \leq m$ ，均有  $d_i \geq n/2$ 。

特殊性质 C： $l = 1$ 。