

## SBSWAP: 交换子树

## 题目描述

你将会得到一个节点有权重的树  $N$ ，每个节点标号为 1 到  $N$ ，而根节点的标号是 1。你的目标是处理  $Q$  个操作，每个操作都是下面三种操作中的一种：

- `getSum(u)`：返回  $u$  的子树的所有节点权值的和；
- `add(u, x)`：把所有  $u$  的子树的节点（包括  $u$ ）的权值加上  $x$ ；
- `swap(u, v)`：如果  $u$  和  $v$  的子树不相交，那么交换这两棵子树。

## 输入格式

输入数据的第一行包含两个空格分隔的整数  $N$  和  $Q$ ，依次代表节点的数量和操作的数量。

输入数据的第二行包含  $N$  个空格分隔的整数  $w_1, w_2, \dots, w_N$ ，其中  $w_i$  表示开始时节点  $i$  的权值。

接下来  $N - 1$  行，每行将会包含两个空格分隔的整数  $u$  和  $v$ ，表示节点  $u$  和  $v$  之间有一条边。

接下来  $Q$  行中的第  $i$  行表示第  $i$  个操作。每行以一个整数  $qtype$  开始，表示操作的种类。如果  $qtype$  是 1，接下来将会有有一个整数，代表操作中的参数。如果  $qtype$  是 2 或 3，接下来将会有两个空格分隔的整数，代表操作中的参数。

## 输出格式

对于每个  $qtype$  为 1 的操作，输出一行，包含一个整数，为询问的结果。而对于每个  $qtype$  为 3 的操作，如果两个子树相交，输出 -1。

## 数据范围和子任务

- $1 \leq v, u \leq N$
- $1 \leq x \leq 10^5$
- $1 \leq w_i \leq 10^5$

## 子任务 1 (10 分)：

- $1 \leq N \leq 1000$
- $1 \leq Q \leq 1000$

## 子任务 2 (20 分)：

- $1 \leq N \leq 10^5$
- $1 \leq Q \leq 10^5$
- 不存在  $qtype$  为 3 的询问
- 时间限制为 3 秒

## 子任务 3 (70 分)：

- $1 \leq N \leq 10^5$
- $1 \leq Q \leq 10^5$
- 时间限制为 4 秒

## 样例数据

## 输入

```
10 5
1 1 1 1 1 1 1 1 1 1
1 2
1 3
1 8
3 4
```

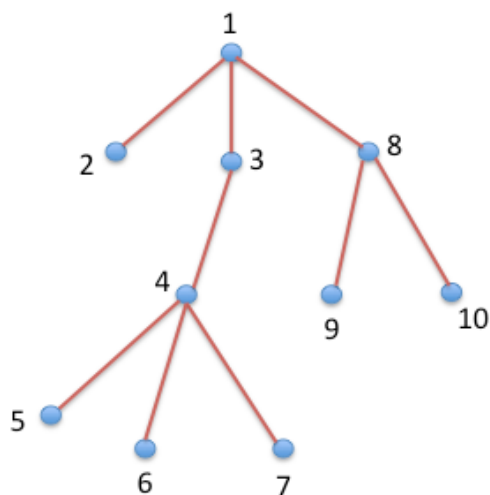
```
8 9
8 10
4 5
4 6
4 7
2 8 1
1 3
3 4 8
1 3
3 1 2
```

### 输出

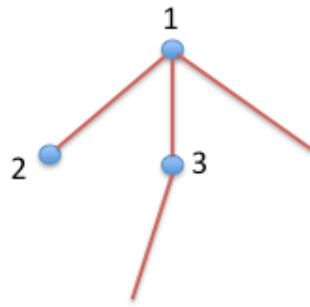
```
5
7
-1
```

### 样例解释

输入的树有 10 个节点，而一开始他们的权重都是 1。下面是初始树的图：

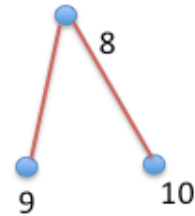
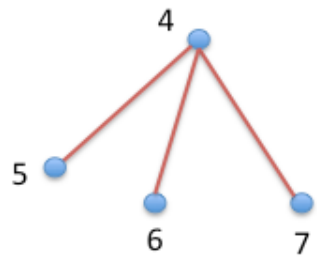


我们有 5 个操作要处理，第一个是将节点 8 的子树的所有节点的权值加上 1。第二个是询问节点 3 的所有子树节点的权值和，因为这一子树包含 5 个节点，权值均为初始值 1，因此答案是 5。第三个操作将交换节点 4 和 8 的子树，其过程如图：

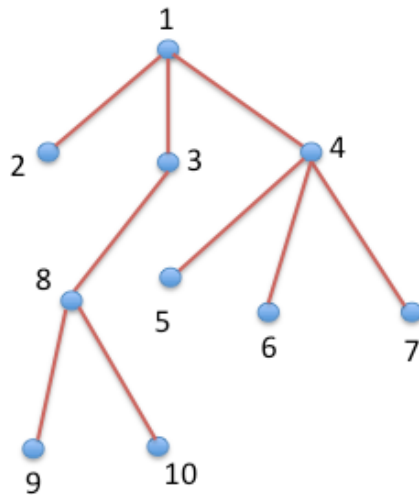


取出以4为根的子树

取出以8为根的子树



交换两棵子树后



因为两棵子树没有相交，这一交换可以进行。第4个操作询问节点3的所有子树节点的权值和，鉴于一些节点已经变换了位置，此时节点3的子树包含一个权值为1的节点和三个权值为2的节点，因此结果为7。最后一个操作试图交换节点1和节点2的子树，但由于这两棵子树相交，对这个操作返回-1。

时间限制

2 ~ 5 秒

---

Problem Setter: Pawel Kacprzak  
Problem Tester: Misha Chorniy  
Translated by: Hu Zecong