

题目名称	软件	最大后缀值个数	树	魔塔
程序名称	software	suffix	tree	tower
输入文件名	software.in	suffix.in	tree.in	tower.in
输入文件名	software.out	suffix.out	tree.out	tower.out
时限	1 秒	1 秒	2 秒	1 秒
内存限制	128MB	128MB	500MB	500MB,
测试点	10	10	20	10

## 软件

### 【题目描述】

一个软件开发公司同时要开发两个软件，并且要同时交付给用户，现在公司为了尽快完成这一任务，将每个软件划分成  $m$  个模块，由公司里的技术人员分工完成，每个技术人员完成同一软件的不同模块的所用的天数是相同的，并且是已知的，但完成不同软件的一个模块的时间是不同的，每个技术人员在同一时刻只能做一个模块，一个模块只能由一个人独立完成而不能由多人协同完成。一个技术人员在整个开发期内完成一个模块以后可以接着做任一软件的任一模块。写一个程序，求出公司最早能在什么时候交付软件。

### 【输入格式】

第一行包含两个由空格隔开的整数  $n$  和  $m$ 。

接下来的  $n$  行每行包含两个用空格隔开的整数  $d_1$  和  $d_2$ ， $d_1$  表示该技术人员完成第一个软件中的一个模块所需的天数， $d_2$  表示该技术人员完成第二个软件中的一个模块所需的天数。

### 【输出格式】

仅有一行包含一个整数  $d$ ，表示公司最早能于  $d$  天后交付软件。

### 【输入样例】

```
3 20
1 1
2 4
1 6
```

### 【输出样例】

```
18
```

样例解释：

最快的方案是第一个技术人员完成第二个软件的 18 个模块，用时 18 天，第三个技术人员完成第一个软件的 18 个模块，用时 18 天，其余的模块由第二个技术人员完成，用时 12 天，做完所有模块需要 18 天。如果第一个技术人员完成第二个软件的 17 个模块，第三个技术人员完成第一个软件的 17 个模块，其余的模块由第二个技术人员完成，需要用时 18 天，做完所有模块仍然需要 18 天，所以少于 18 天不可能做完所有模块。

### 【数据范围】

40%的数据， $1 \leq n \leq 20$ ,  $1 \leq m \leq 20$

100%的数据， $1 \leq n \leq 100$ ,  $1 \leq m \leq 100$ ,  $1 \leq d_1, d_2 \leq 100$

## 最大后缀值个数

### 【问题描述】

一棵根节点为1的树，点有点权，求出根节点到每个点的有向链上的后缀最大值的个数。

形式化地，假设从1到点u的有向路径(含端点)上的节点分别为 $p_1, p_2, \dots, p_k$ ，你需要求出满足

$$1 \leq i \leq k, \text{val}_{p_i} = \max_{i \leq j \leq k} \text{val}_{p_j}$$

的i的个数。

### 【输入格式】

第一行一个正整数  $n$ ，树的大小。

第二行 $n-1$ 个正整数，编号从2~ $n$ 的节点的父亲。

第三行 $n$ 个正整数，每个节点的权值 $v_i$ 。

### 【输出格式】

一行 $n$ 个正整数，每个节点的答案。

### 【样例输入1】

```
5
1 1 2 2
5 2 1 4 3
```

### 【样例输出1】

```
1 2 2 2 2
```

### 【样例输入2】

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

### 【样例输出2】

```
1 1 1 1 1 1 1 1 1 1
```

### 【数据范围】

对于10% 的数据， $2 \leq n \leq 10$  。

对于另外40% 的数据， $x$  的父节点是 $x-1$  。

对于100% 的数据， $2 \leq n \leq 10^6$ ， $1 \leq v_i \leq 10^9$  。

树

**【题目描述】**

小 X 的同学的座位呈一个树状结构,由  $n$  个点和  $n-1$  条边组成,1 号结点为根。  
当小 X 的一个同学无视 小 X 后,一个单位时间后周围的同学也会无视小 X。  
有些时候小 X 为了吸引同学们的注意,会开启鬼畜模式,这时所有同学无视小 X 的状态都会消失,恢复正常,并且这之后的状态不会被之前影响。  
小 X 想知道某个同学是否无视了他,于是他找到了你,请你帮帮他。

**【输入描述】**

两个数  $n, m$ , 代表同学数与询问个数。  
第 2 到第  $n$  行每行 2 个整数  $u_i, v_i$ , 代表这两个同学座位之间有边相连。  
接下来  $m$  行是两个数  $opt, x$ , 代表操作编号与同学编号。  
如果  $opt=1$ , 代表同学  $x$  开始无视小 X 了。  
如果  $opt=2$ , 代表小 X 开始鬼畜了,所有同学无视小 X 的状态都消失了,请忽略该操作的  $x$ 。  
如果  $opt=3$ , 代表小 X 想询问  $x$  同学是否无视他。  
其中  $opt=1, 2, 3$  的操作会占用一个单位时间,且操作 1 后的下一个单位时间开始时无视状态开始扩散,操作 3 是在该单位时间恰好结束时询问。

**【输出描述】**

对于每个  $opt=3$  的询问,若该同学无视小 X, 输出 `wrxcsd`, 否则输出 `orzFsYo`。(不需要输出引号)

**【输入样例 1】**

```
5 4
1 2
1 3
2 4
2 5
1 2
3 2
2 4
3 4
```

**【输出样例 1】**

```
wrxcsd
orzFsYo
```

说明

样例 1 中,第一次操作后同学 2 的无视小 X,所以询问时输出 `wrxcsd`。经过操作 2 后,同学 1, 4, 5 均无视小 X,操作 3 将所有同学恢复正常,故操作 4 输出 `orzFsYo`。

**【输入样例 2】**

```
10 10
2 1
```

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

**【输出样例 2】**

orzFsYo  
wrxcsd  
wrxcsd

**【数据范围】**

对于 20%的数据：  $n, m \leq 1000$

对于另 10%的数据：  $u_i = 1$

对于另 10%的数据，保证  $u_i = i, v_i = i + 1$

对于另 15%的数据： 保证  $u_i = \lfloor (i+1)/2 \rfloor, v_i = i + 1$

对于 100%的数据：  $n, m \leq 10^5, 1 \leq \text{opt} \leq 3, 1 \leq x \leq n$

## 魔塔

### 【题目描述】

小 X 的同学被大魔王抓走了，小 X 为了就他的同学，前往攻打魔塔。

魔塔为一棵树，小 X 初始在一号点。

小 X 有攻击，防御，血量三个属性。

除一号点外每个点都有魔物防守，魔物也有攻击，防御，血量三个属性。

每个怪物后面都守着一些蓝宝石，获得 1 蓝宝石可增加 1 防。

小 X 具有突袭属性，所以遇到魔物后会率先发动攻击，然后小 X 和魔物轮换地攻击对方。

一个角色被攻击一次减少的血量是对方的攻击减去自己的防御。

当一个角色的血量小于等于 0 时，他就会死亡。

当小 X 第一次到达某个节点时会与这个节点的魔物发生战斗。

当一个魔物死亡后，这个魔物所在的节点就不会再产生新的魔物。

现在小 X 想知道他打死魔塔的所有魔物后的最大血量。

### 【输入描述】

第一行一个  $n$  代表节点数。

随后  $n-1$  行，每行两个数  $i, j$ ，表示  $i$  与  $j$  节点有边相连。

随后一行，三个数，依次为勇士的血量、攻击、防御。

随后  $n-1$  行，每行四个数，依次为怪物的血量、攻击、防御，和其守着的蓝宝石数量。

### 【输出描述】

一个数，代表最大血量。如果小 X 在打死魔塔的所有魔物之前就已经死亡了，则输出  $-1$ 。

### 【输入样例】

```
6
1 2
1 3
1 4
4 5
5 6
50000 10 0
35 54 2 4
25 55 3 5
21 51 4 5
20 64 5 3
43 64 6 1
```

### 【输出样例】

```
48901
说明
打怪的顺序依次为 4, 3, 5, 2, 6
可以证明不存在更优的方案。
```

**【数据范围】**

对于 10%的数据:  $n \leq 15$ , 树

另有 10%的数据:  $n \leq 10^5$ , 只存在边  $(1, i)$

另有 10%的数据:  $n \leq 10^5$ , 只存在边  $(i-1, i), (1, i)$

另有 30%的数据:  $n \leq 10^3$ , 树

对于 100%的数据:  $n \leq 10^5$ , 树

对于 100%的数据: 有小  $X$  血量  $< 5 \times 10^{18}$ , 攻击 = 2000, 盔甲防御 = 0。怪物血量为  $3000 \sim 10^6$ , 攻击  $5 \times 10^5 \sim 7 \times 10^5$ , 防御  $\leq 1000$ , 打完一只怪后获得的蓝宝石数量为 1 至 5