

Problem A. 原题识别 (old.c/cpp/pas)

Input file: old.in
 Output file: old.out
 Time limit: 10 seconds
 Memory limit: 512 megabytes

“人肉题库”小 Q 刷题非常勤奋，题量破万。每当有人拿题目请教他时，小 Q 总能在 1 秒内报出这是哪个 OJ 的哪道题。因此，小 Q 是被当作“原题搜题机”一样的存在。

有一天，小 Q 来到了一棵 n 个节点的有根树下，这棵树的根节点为 1 号点，且每个节点都印着一道题目。凭借超大的题量，小 Q 迅速识别出了每道题的来源，并发现有些题目被搬运了好多次。他把每个节点的题目都做了一个分类，第 i 个节点的题目对应的题目种类为 a_i ，当且仅当 $a_i = a_j$ 时， i 点和 j 点的题目来源是相同的。

同一道题目做多次除了增加 AC 数以外，对本身的水平没有任何提高。为了调查这棵树的题目质量，小 Q 会不断提出以下两种询问共 m 次：

- 1 $x\ y$: 如果将 x 点到 y 点的最短路径上的所有点 (包括 x 和 y) 对应的题目都做一遍，那么一共可以做到多少道本质不同的题目？
- 2 $A\ B$: 如果在 A 点到根的最短路径上 (包括 A 点和根) 等概率随机选择一个点 x ，在 B 点到根的最短路径上 (包括 B 点和根) 等概率随机选择一个点 y ，那么询问 1 $x\ y$ 的答案期望是多少？

定义 cnt_x 表示 x 点到根最短路径上的节点个数，因为小 Q 不喜欢分数，而且第 2 类询问的答案一定可以表示成 $\frac{ans}{cnt_A \times cnt_B}$ 的形式，你只需要告诉他 ans 的值就可以了。

识别这些题目消耗了小 Q 太大的精力，他没有办法自己去计算这些简单的询问的答案。请写一个程序回答小 Q 的所有 m 个问题。

Input

第一行包含一个正整数 T ，表示测试数据的组数。

每组数据第一行包含 5 个正整数 n, p, SA, SB, SC ，其中 n 表示树的节点个数。

为了在某种程度上减少输入量，树边和每个点的题目种类 $a[]$ 将由以下代码生成：

```

1 unsigned int SA, SB, SC;
2 unsigned int rng61(){
3     SA ^= SA << 16;
4     SA ^= SA >> 5;
5     SA ^= SA << 1;
6     unsigned int t = SA;
7     SA = SB;
8     SB = SC;
9     SC ^= t ^ SA;
10    return SC;
11 }
```

```

12 void gen(){
13     scanf("%d%d%u%u%u", &n, &p, &SA, &SB, &SC);
14     for(int i = 2; i <= p; i++)
15         addedge(i - 1, i);
16     for(int i = p + 1; i <= n; i++)
17         addedge(rng61() % (i - 1) + 1, i);
18     for(int i = 1; i <= n; i++)
19         a[i] = rng61() % n + 1;
20 }

```

第二行包含一个正整数 m ，表示询问次数。

接下来 m 行，每行 3 个正整数，形式为 $1\ x\ y$ 或者 $2\ A\ B$ ，依次表示每个询问。

Output

对于每组数据，输出 m 行，每行一个整数，依次回答每个询问。

Examples

old.in	old.out
2	1
5 3 10000 12345 54321	5
3	1
1 2 3	4
2 1 3	34
1 3 2	61
10 6 23456 77777 55555	45
5	3
1 1 10	
2 3 5	
2 7 5	
2 5 4	
1 8 6	

Notes

- $1 \leq T \leq 3$, $2 \leq p \leq n \leq 100000$, $1 \leq m \leq 200000$,
- $10000 \leq SA, SB, SC \leq 1000000$, $1 \leq x, y, A, B \leq n$ 。

子任务 1 (30 分): 只含第 1 类询问。

子任务 2 (30 分): 满足 $p = n$ 。

子任务 3 (40 分): 没有任何附加的限制。