

全国青少年信息学奥林匹克（NOI2017）

山东省队第二轮选拔赛

第一试

考试时间：2017 年5 月17 日8：00—13：00

对于C/C++程序，最终测试将开启 -O2 优化

Problem A. 龙与地下城 (dnd.c/cpp/pas)

Input file: dnd.in
 Output file: dnd.out
 Time limit: 4 seconds
 Memory limit: 512 megabytes
 Feedback: **Special Judge**

小Q同学是一个热爱学习的人，但是他最近沉迷于各种游戏，龙与地下城就是其中之一。

在这个游戏中，很多场合需要通过掷骰子来产生随机数，并由此决定角色未来的命运，因此骰子堪称该游戏的标志性道具。

骰子也分为许多种类，比如4面骰、6面骰、8面骰、12面骰、20面骰，其中20面骰用到的机会非常多。当然，现在科技发达，可以用一个随机数生成器来取代真实的骰子，所以这里认为骰子就是一个随机数生成器。

在战斗中，骰子主要用来决定角色的攻击是否命中，以及命中后造成的伤害值。举个例子，假设现在已经确定能够命中敌人，那么 YdX （也就是掷出 Y 个 X 面骰子之后所有骰子显示的数字之和）就是对敌人的基础伤害。在敌人没有防御的情况下，这个基础伤害就是真实伤害。

众所周知，骰子显示每个数的概率应该是相等的，也就是说，对于一个 X 面骰子，显示 $0, 1, 2, \dots, X-1$ 中每一个数字的概率都是 $\frac{1}{X}$ 。

更形式地说，这个骰子显示的数 W 满足离散的均匀分布，其分布列为

W	0	1	2	...	$X-1$
P	$\frac{1}{X}$	$\frac{1}{X}$	$\frac{1}{X}$...	$\frac{1}{X}$

除此之外还有一些性质

- W 的一阶原点矩(期望)为 $\nu_1(W) = E(W) = \sum_{i=0}^{X-1} iP(W=i) = \frac{X-1}{2}$
- W 的二阶中心矩(方差)为 $\mu_2(W) = E((W - E(W))^2) = \sum_{i=0}^{X-1} (i - E(W))^2 P(W=i) = \frac{X^2-1}{12}$

言归正传，现在小Q同学面对着一个生命值为 A 的没有防御的敌人，能够发动一次必中的 YdX 攻击，显然只有造成的伤害不少于敌人的生命值才能打倒敌人。但是另一方面，小Q同学作为强迫症患者，不希望出现overkill，也就是造成的伤害大于 B 的情况，因此只有在打倒敌人并且不发生overkill的情况下小Q同学才会认为取得了属于他的胜利。

因为小Q同学非常谨慎，他会进行10次模拟战，每次给出敌人的生命值 A 以及overkill的标准 B ，他想知道此时取得属于他的胜利的概率是多少，你能帮帮他吗？

Input

第一行是一个正整数 T ，表示测试数据的组数，

对于每组测试数据，

第一行是两个整数 X, Y ，分别表示骰子的面数以及骰子的个数，

接下来10行，每行包含两个整数 A, B ，分别表示敌人的生命值 A 以及overkill的标准 B 。

Output

对于每组测试数据，输出10行，对每个询问输出一个实数，要求绝对误差不超过0.013579，

也就是说，记输出为 a ，答案为 b ，若满足 $|a - b| \leq 0.013579$ ，则认为输出是正确的。

Examples

dnd.in	dnd.out
1	0.000002
2 19	0.000038
0 0	0.000364
0 1	0.002213
0 2	0.009605
0 3	0.031784
0 4	0.083534
0 5	0.179642
0 6	0.323803
0 7	0.500000
0 8	
0 9	

Notes

对于100%的数据， $T \leq 10$ ， $2 \leq X \leq 20$ ， $1 \leq Y \leq 200000$ ， $0 \leq A \leq B \leq (X - 1)Y$ ，保证满足 $Y > 800$ 的数据不超过2组。

测试点编号	X	Y	
1	≤ 20	≤ 40	$X^Y \leq 10000000$
2,3,4	≤ 20	≤ 1600	
5,6,7,8,9,10	≤ 20	≤ 8000	
11,12	$= 2$	≤ 200000	
13,14,15,16,17,18,19,20	≤ 20	≤ 200000	

Problem B. 苹果树 (apple.c/cpp/pas)

Input file: apple.in
Output file: apple.out
Time limit: 5 seconds
Memory limit: 512 megabytes

夏天近了，又到了恋爱的季节，小Q家门前的苹果树上结满了红红圆圆的苹果。

这株苹果树是一个有着 n 个结点的有根树，其中结点被依次编号为1至 n 。1号结点为根，其余每一个结点的父结点一定是某个编号较小的结点。每一个结点上都有一些苹果，第 i 个结点上有 a_i ($a_i > 0$)个苹果，每取走其中一个苹果就可以得到 v_i ($v_i > 0$)的幸福度（若在这个结点取走 $k \leq a_i$ 个苹果，则可以收获 kv_i 的幸福度）。如果在一个结点取走了至少一个苹果，则必须要在其父结点处取走至少一个苹果。

现在，给定正整数 k ，请从树上取走若干苹果。如果总计取走了 t 个苹果，且所有取了至少一个苹果的那些结点的最大深度为 h （这里规定根结点的深度为1），则要求 $t - h \leq k$ 。问最大可以收获多少的幸福度？（这些幸福度全都归属于恋爱中的小Q。）

Input

本题有多组测试数据，输入的第一行给定整数 Q 表示有 Q 组数据。之后依次给出 Q 组数据。

对于每一组数据来说，第一行包含两个整数 n 和 k 。

之后 n 行，每行给出三个整数，描述了每一个结点。其中第 i 行的第一个整数给出了 i 的父结点标号（如果 $i = 1$ ，则其父结点为0），第二个整数为 a_i ，第三个整数为 v_i 。

Output

输出一共有 Q 行，对应了 Q 组数据。

对于每一组数据，输出一个整数，表示最大可以收获的幸福度。

Examples

apple.in	apple.out
2	15
5 1	316
0 1 1	
1 1 1	
1 1 3	
2 1 10	
3 1 4	
9 15	
0 1 1	
1 7 2	
2 5 10	
1 3 1	
4 3 17	
4 3 18	
4 4 19	
1 1 1	
8 1 100	

Notes

有10%的数据，满足 $nk \leq 3000000$ 且给定的树的高度为2。

有20%的数据，满足 $nk \leq 25000000$ 且给定的树的高度为2。

有20%的数据，满足 $nk \leq 25000000$ 且所有 a_i 均为1。

还有20%的数据，满足 $nk \leq 3000000$ ，没有上述额外限制。

对于100%的数据，满足 $1 \leq Q \leq 5$, $1 \leq n \leq 20000$, $1 \leq k \leq 500000$, $1 \leq nk \leq 25000000$, $1 \leq a_i \leq 10^8$, $1 \leq v_i \leq 100$ 。

Problem C. 切树游戏 (cut.c/cpp/pas)

Input file: cut.in
Output file: cut.out
Time limit: 3 seconds
Memory limit: 512 megabytes

小Q是一个热爱学习的人，他经常去维基百科学习计算机科学。

就在刚才，小Q认真地学习了一系列位运算符，其中按位异或的运算符 \oplus 对他影响很大。按位异或的运算符是双目运算符。按位异或具有交换律，即 $i \oplus j = j \oplus i$ 。

他发现，按位异或可以理解成被运算的数字的二进制位对应位如果相同，则结果的该位置为0，否则为1，例如： $1(01) \oplus 2(10) = 3(11)$ 。

他还发现，按位异或可以理解成参与运算的数字的每个二进制位都进行了不进位的加法，例如： $3(11) \oplus 3(11) = 0(00)$ 。

现在小Q有一棵 n 个结点的无根树 T ，结点依次编号为1到 n ，其中结点 i 的权值为 v_i 。

定义一棵树的价值为它所有点的权值的异或和，一棵树 T 的连通子树就是它的一个连通子图，并且这个图也是一棵树。

小Q想要在这棵树上玩切树游戏，他会不断做以下两种操作：

- Change $x y$ 将编号为 x 的结点的权值修改为 y 。
- Query k 询问有多少棵 T 的非空连通子树，满足其价值恰好为 k 。

小Q非常喜(bu)欢(hui)数学，他希望你能快速回答他的问题，你能写个程序帮帮他吗？

Input

第一行包含两个正整数 n, m ，分别表示结点的个数以及权值的上限。

第二行包含 n 个非负整数 v_1, v_2, \dots, v_n ，分别表示每个结点一开始的权值。

接下来 $n - 1$ 行，每行包含两个正整数 a_i, b_i ，表示有一条连接 a_i 和 b_i 的无向树边。

接下来一行包含一个正整数 q ，表示小Q操作的次数。

接下来 q 行每行依次表示每个操作。

Output

输出若干行，每行一个整数，依次回答每个询问。因为答案可能很大，所以请对10007取模输出。

Examples

cut.in	cut.out
4 4	3
2 0 1 3	3
1 2	2
1 3	3
1 4	2
12	4
Query 0	2
Query 1	3
Query 2	
Query 3	
Change 1 0	
Change 2 1	
Change 3 3	
Change 4 1	
Query 0	
Query 1	
Query 2	
Query 3	

Notes

对于100%的数据， $1 \leq a_i, b_i, x \leq n, 0 \leq v_i, y, k < m$ ，修改操作不超过10000个。

测试点编号	n	m	q	约定
1,2,3,4	≤ 2000	$= 64$	≤ 64	不存在修改操作
5,6,7,8	≤ 30000	$= 8$	≤ 30000	$a_i = i, b_i = i + 1$
9,10	≤ 30000	$= 128$	≤ 30000	$a_i = i, b_i = i + 1$
11,12,13,14,15	≤ 30000	$= 4$	≤ 30000	无
16,17,18,19,20	≤ 30000	$= 128$	≤ 30000	无