

Clariss' Contest # 2

Day 2

By Claris

September 3, 2016

题目名称	Divisors	Market	Dash Speed
输入文件名	div.in	market.in	speed.in
输出文件名	div.out	market.out	speed.out
每个测试点时限	1s	1s	1s
测试点数目	10	10	10
每个测试点分值	10	10	10
内存限制	128MB	128MB	128MB
是否有部分分	否	否	否
题目类型	传统	传统	传统

Problem A. Divisors(div.c/cpp/pas)

Input file: div.in
Output file: div.out
Time limit: 1 seconds
Memory limit: 128 megabytes

给定 m 个不同的正整数 a_1, a_2, \dots, a_m , 请对 0 到 m 每一个 k 计算, 在区间 $[1, n]$ 里有多少正整数是 a 中恰好 k 个数的约数。

Input

第一行包含两个正整数 n, m , 分别表示区间范围以及 a 数组的大小。

第二行包含 m 个不同的正整数 a_1, a_2, \dots, a_m , 表示 a 数组。

Output

输出 $m+1$ 行, 每行一个整数, 其中第 i 行输出 $k=i$ 的答案。

Examples

div.in	div.out
10 3 4 6 7	4 4 1 1
5 1 8	2 3

Notes

测试点编号	m	n, a_i
1	$= 5$	≤ 1000
2	$= 50$	≤ 1000
3	$= 200$	≤ 1000
4	$= 1$	$\leq 10^9$
5	$= 1$	$\leq 10^9$
6	$= 1$	$\leq 10^9$
7	$= 200$	$\leq 10^9$
8	$= 200$	$\leq 10^9$
9	$= 200$	$\leq 10^9$
10	$= 200$	$\leq 10^9$

Problem B. Market(market.c/cpp/pas)

Input file: `market.in`
Output file: `market.out`
Time limit: 1 seconds
Memory limit: 128 megabytes

在比特镇一共有 n 家商店，编号依次为 1 到 n 。每家商店只会卖一种物品，其中第 i 家商店的物品单价为 c_i ，价值为 v_i ，且该商店开张的时间为 t_i 。

Byteasar 计划进行 m 次购物，其中第 i 次购物的时间为 T_i ，预算为 M_i 。每次购物的时候，Byteasar 会在每家商店购买最多一件物品，当然他也可以选择什么都不买。如果购物的时间早于商店开张的时间，那么显然他无法在这家商店进行购物。

现在 Byteasar 想知道，对于每个计划，他最多能购入总价值多少的物品。请写一个程序，帮助 Byteasar 合理安排购物计划。

注意：每次所花金额不得超过预算，预算也不一定要花完，同时预算不能留给其它计划使用。

Input

第一行包含两个正整数 n, m ，表示商店的总数和计划购物的次数。

接下来 n 行，每行三个正整数 c_i, v_i, t_i ，分别表示每家商店的单价、价值以及开张时间。

接下来 m 行，每行两个正整数 T_i, M_i ，分别表示每个购物计划的时间和预算。

Output

输出 m 行，每行一个整数，对于每个计划输出最大可能的价值和。

Examples

market.in	market.out
5 2	10
5 5 4	12
1 3 1	
3 4 3	
6 2 2	
4 3 2	
3 8	
5 9	

第一个计划可以在商店 2,3,5 各购买一件物品，总花费为 $1 + 3 + 4 = 8$ ，总价值为 $3 + 4 + 3 = 10$ 。

第二个计划可以在商店 1,2,3 各购买一件物品，总花费为 $5 + 1 + 3 = 9$ ，总价值为 $5 + 3 + 4 = 12$ 。

Notes

对于 100% 的数据, $1 \leq t_i, T_i \leq n$ 。

测试点编号	n	m	c_i, M_i	v_i	t_i, T_i
1	= 10	= 5	≤ 10	≤ 10	≤ 10
2	= 20	= 10	≤ 100	≤ 100	≤ 20
3	= 100	= 1	≤ 100	≤ 100	= 1
4	= 200	= 1	≤ 200	≤ 200	= 1
5	= 150	= 100000	≤ 150	≤ 150	≤ 150
6	= 300	= 100000	≤ 300	≤ 300	≤ 300
7	= 20	= 100000	$\leq 10^9$	≤ 300	≤ 20
8	= 200	= 100000	$\leq 10^9$	≤ 200	≤ 200
9	= 300	= 100000	$\leq 10^9$	≤ 300	≤ 300
10	= 300	= 100000	$\leq 10^9$	≤ 300	≤ 300

Problem C. Dash Speed(speed.c/cpp/pas)

Input file: speed.in
Output file: speed.out
Time limit: 1 seconds
Memory limit: 128 megabytes

比特山是比特镇的飙车圣地。在比特山上一共有 n 个广场，编号依次为 1 到 n ，这些广场之间通过 $n - 1$ 条双向车道直接或间接地连接在一起，形成了一棵树的结构。

因为每条车道的修建时间以及建筑材料都不尽相同，所以可以用两个数字 l_i, r_i 量化地表示一条车道的承受区间，只有当汽车以不小于 l_i 且不大于 r_i 的速度经过这条车道时，才不会对路面造成伤害。

Byteasar 最近新买了一辆跑车，他想要在比特山飙一次车。Byteasar 计划选择两个不同的点 S, T ，然后在它们树上的最短路径上行驶，且不对上面任意一条车道造成伤害。

Byteasar 不喜欢改变速度，所以他会告诉你他的车速。为了挑选出最合适的车速，Byteasar 一共会向你询问 m 次。请帮助他找到一条合法的道路，使得路径上经过的车道数尽可能多。

Input

第一行包含两个正整数 n, m ，表示广场的总数和询问的总数。

接下来 $n - 1$ 行，每行四个正整数 u_i, v_i, l_i, r_i ，表示一条连接 u_i 和 v_i 的双向车道，且承受区间为 $[l_i, r_i]$ 。

接下来 m 行，每行一个正整数 q_i ，分别表示每个询问的车速。

Output

输出 m 行，每行一个整数，其中第 i 行输出车速为 q_i 时的最长路径的长度，如果找不到合法的路径则输出 0。

Examples

speed.in	speed.out
5 3	0
3 2 2 4	2
1 5 2 5	3
4 5 2 2	
1 2 3 5	
1	
2	
3	

当车速为 1 时，不存在合法的路径。

当车速为 2 时，可以选择 1-5-4 这条路径，长度为 2。

当车速为 3 时，可以选择 3-2-1-5 这条路径，长度为 3。

Notes

对于 100% 的数据, $1 \leq u_i, v_i, q_i \leq n, 1 \leq l_i \leq r_i \leq n$ 。

测试点编号	n	m	约定
1	= 5	= 5	无
2	= 20	= 20	无
3	= 50000	= 50000	$l_i = 1$, 且 $u_i = i, v_i = i + 1$
4	= 70000	= 70000	$l_i = 1$, 且 $u_i = i, v_i = i + 1$
5	= 50000	= 50000	$u_i = i, v_i = i + 1$
6	= 70000	= 70000	$u_i = i, v_i = i + 1$
7	= 50000	= 50000	$l_i = 1$
8	= 70000	= 70000	$l_i = 1$
9	= 50000	= 50000	无
10	= 70000	= 70000	无