机房里除了太阳们,还有一位鸽王。有一天,鸽王来到机房,发现桌上有不知道哪个蒟蒻放上的问题:

现在有一个集合 $K = \{0, 1, 2, \dots, k-1\}$,定义一种运算 \oplus 。 $\forall a, b \in K, \exists c \in K$,使得 $a \oplus b = c$ 。

没有交换律当然很不爽,因此这种运算自然有交换律,即 $\forall a,b \in K, a \oplus b = b \oplus a$ 。

没有结合律当然很不爽,因此这种运算自然有结合律,即 $\forall a,b,c \in K, (a \oplus b) \oplus c = a \oplus (b \oplus c)$ 。

没有对任何运算的分配律当然很不爽,但是 A 国的人很累了,所以不想再做任何保证。

A 国有 n 个城市,有 n-1 条道路连通所有城市,每条道路上有权 $c \in K$ 。有一个旅者要从一个城市沿着最短路径走到另一个城市,一开始他手上的数是 0,当他手上的数为 a 时,经过一条权为 c 的道路后手上的数会变成 $a \oplus c$ 。

A 国的人认为排列是平均的,而平平稳稳才是生活的真谛。因此对任意的 i, $\{i \oplus j | j \in K\} = K$,也就是说 i 和所有的 j 进行 \oplus 运算后是一个 0 到 k-1 的排列。

A 国的人认为人是生来平等的,所以即便城市之间会有矛盾,每座城市也只会与至多一座城市互相敌对。 当然有的城市热爱和平,所以会宽容地对待所有其他城市的人们。如果两座城市互相敌对,那么旅者就会 认为同时包含这两座城市的路径是不安全的。1 号城市是首都,所以每座城市都不会敌对其到首都路径上 的城市的地步,因为这样就不能说对方坏话了。现在旅者想要知道,对于所有的安全的路径,他最终手上 的数是多少。因为路径很多,所以你只需要输出所有数的和。注意 u->v 和 v->u 是两条不同的路径。 鸽王非常强,自然不会把时间花在这种水题上。因此他就把任务交给了你,如果不会做的话,他就只能鸽 了这只蒟蒻了。

Input

第一行一个字符串 str,表示数据类型。

第二行一个正整数 k,表示集合 K 的大小,保证 k > 1。

接下来 k 行每行 k 个数, 第 i 行第 j 个数表示 $i \oplus j$ 的值, 数据保证满足上文所提到的运算律。

接下来一行一个正整数 n,表示城市的总数。

接下来 n-1 行, 每行三个数 u,v,c 表示在城市 u 和城市 v 之间有一条权为 c 的道路。

接下来一行 n 个数,第 i 个数表示城市 i 的敌对城市 x_i ,若 $x_i = 0$ 表示第 i 座城市非常热爱和平,否则保证有 $x_{x_i} = i$ 。

Output

输出共一行,表示所有路径的旅者最终手上的数之和。

Examples

xor.in	xor.out		
F	6		
2			
0 1			
1 0			
4			
1 2 1			
2 3 1			
2 4 0			
0 0 4 3			
见 $xor_example_1 - 4.in$	见 $xor_example_1 - 4.ans$		

下。

城市编号	1	2	3	4
1	0	1	0	1
2	1	0	1	0
3	0	1	0	-
4	1	0	_	0

表中-表示路径上存在城市互相敌对,否则第 i 行第 j 列就是从城市 i 走到城市 j 后手上剩下的数,显然答案为 6。

Notes

测试点编号	k	n	数据类型	测试点编号	k	n	数据类型
1	≤ 100	≤ 300	A	14	≤ 10	$\leq 10^5$	CD
2	≤ 100	≤ 300	BD	15	≤ 10	$\leq 10^{5}$	Е
3	≤ 100	≤ 300	E	16	≤ 100	≤ 30000	D
4	≤ 100	≤ 1000	В	17	≤ 100	≤ 50000	E
5	≤ 100	≤ 1000	D	18	≤ 100	≤ 70000	В
6	≤ 100	≤ 1000	Е	19	≤ 100	≤ 80000	D
7	≤ 100	≤ 2000	В	20	≤ 100	≤ 90000	E
8	≤ 100	≤ 3000	E	21	≤ 128	$\leq 10^{5}$	В
9	≤ 100	≤ 4000	E	22	≤ 128	$\leq 1.3 * 10^5$	E
10	≤ 100	≤ 5000	CD	23	≤ 128	$\leq 1.5 * 10^5$	BD
11	≤ 50	≤ 10000	В	24	≤ 128	$\leq 2 * 10^5$	E
12	≤ 50	≤ 20000	D	25	≤ 128	$\leq 2 * 10^5$	E
13	≤ 50	≤ 30000	A				

数据类型	A	В	С	D	Е
数据性质	树是一条链	$i \oplus j = i \wedge j$	$i \oplus j = (i+j) \bmod k$	$x_i = 0$	无