

中文题目名称	签到题不需要题目名称	木积	远游	蒟蒻座蒟蒻星
英文题目名称	easy	multiply	walk	konjac
源程序文件名	easy.cpp	multiply.cpp	walk.cpp	konjac.cpp
输入文件名	easy.in	multiply.in	walk.in	konjac.in
输出文件名	easy.out	multiply.out	walk.out	konjac.out
每个测试点时限	1.0s	1.0s	2.0s	3.0s
内存上限	256MB	256MB	512MB	512MB
测试点组数	20	20	20	20
每个测试点分值	5	5	5	5
比较方式	全文比较	全文比较	全文比较	全文比较
题目类型	Special Judge	传统	传统	传统
编译命令	-lm	-lm	-lm	-lm

1 签到题不需要题目名称 (easy)

1.1 Description

-警告：题目名称不能为空

“哦好吧我改改……”

-警告：题目名称长度不得超过 10 个汉字

“……”

给定一张 m 条边， n 个顶点组成的无向图 G ，顶点从 1 到 n 依次标号，保证 G 中任意两点均可互相到达。

现需要给 G 中尽量多的边指定方向，构成一张新的混合图 G' ，使得 G' 中任意两点 仍均可达。

1.2 Input

输入的第一行包含两个正整数 n, m 。

接下来的 m 行中，第 i 行包含两个正整数 p_i, q_i ，表示 p_i 和 q_i 之间有一条无向边。

1.3 Output

输出一行，包含一个长为 m 的字符串。

对于输入的第 i 条边 $\langle p_i, q_i \rangle$ ，如果你为它指定的方向为 $p_i \rightarrow q_i$ ，输出的第 i 位

为 1；如果你为它指定的方向为 $q_i \rightarrow p_i$ ，输出的第 i 位为 2；如果你未为它指定方向，输出的第 i 位为 0。

1.4 Sample Input

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

1.5 Sample Output

```
112202121
```

1.6 Sample Explanation

容易发现该方案中被指定方向的边已达最大值（8 条），且新图中的点两两可达。

另一种可行解为 112201212

1.7 Data Range

对于 10% 的数据， $n, m \leq 10$ 。

对于 30% 的数据， $n, m \leq 3000$ 。

对于另外 10% 的数据， $m \leq n$ 。

对于另外 10% 的数据， $m = n(n-1)/2$ 。

对于所有数据， $n, m \leq 1000000$ ， $p_i, q_i \leq n$ ，保证无重边，无自环，保证输入中的图联通。

评测时使用 **Special Judge** 判断你的答案的正确性。

对于你给出的答案 G' ，当且仅当其中有向边的条数达到最大值且图中的点两两可达时，该测试点得满分。否则，该测试点得 0 分。

2 木积 (multiply)

2.1 Description

世界上最良心的出题人 gyy 曾经正在准备一套 NOIP 模拟赛题。因为自身实力极为有限，gyy 无法一个人原创难度恰当的题目作为模拟赛题。于是，经过一番深思熟虑后，gyy 蒯下 n 道题目，并准备从中选出一些题作为模拟赛题。每道题目都有其难度值 a_i 。

gyy 发现，一些题目的组合是优秀的，而另一些题目组合将给模拟赛的整体质量产生负面影响。具体地，对于两道题目的组合，当且仅当其难度值互质时，它们的组合是优秀的。对于一组多个题目，当且仅当其中的题目两两组合都是优秀的时候（即其中题目难度两两互质），这组题目是优秀的。

gyy 准备从这 n 道题目中选出尽可能多的题目，使得这些题目的组合是优秀的。请计算出 gyy 能选择的题目数量的最大值。

2.2 Input

本题一个测试点有多组数据，当且仅当你对于每组数据的输出全部正确时，该测试点得满分。

第一行一个正整数 T ，表示数据组数。

对于每组数据：第一行一个正整数 n ；第二行 n 个正整数，第 i 个数 a_i 表示第 i 道题目的难度值。

2.3 Output

对于每组数据，输出一行一个整数，表示 gyy 能选择的题目数量的最大值。

2.4 Sample Input

```
3
5
6 7 8 9 10
4
699 932 233 466
5
74 52 8 39 37
```

2.5 Sample Output

```
3
1
3
```

2.6 Sample Explanation

对于第一组数据，选 7, 9, 10。（方案不唯一）

对于第二组数据，易发现所有数都是 233 的倍数，所以最多只能选一个。

对于第三组数据，选 8, 39, 37。

2.7 Data Range

对于 10% 的数据， $n \leq 5$ 。

对于 30% 的数据， $n \leq 40$ 。

对于另外 10% 的数据， $a_i \leq 100$ 。

对于另外 10% 的数据， a_i 最多有 2 个质因子。

对于 100% 的数据， $n \leq 1000$ ， $a_i \leq 1000$ ， $T \leq 10$ 。

注意：本题唯一一组大样例涵盖了本题所有出现性质的组合，请妥善利用之。

3 远游 (walk)

3.1 Description

有一条 3 行 N 列的走廊，该走廊被划分为 $3 * N$ 个单位网格。记第 i 行第 j 列的 网格坐标为 (i, j) 。

走廊上有若干障碍。障碍分两种：小障碍长宽均为 1，恰好占据一个单位网格；大障碍长宽均为 2，恰好占据 4 个单位网格。障碍之间不可重叠，每个障碍都必须完全落在 走廊内。同种障碍不可区分（即：无标号）。

zyc 在走廊上游走。设 zyc 当前位置为 (x, y) ，则下一步可以走到 $(x + 1, y)(x - 1, y)(x, y + 1)(x, y - 1)$ 中任一个位置，满足该位置在走廊内且没有障碍。

现在 zyc 在走廊外端（视作第 0 列，可走到第一列任一个无障碍位置）。走廊里的 障碍等概率随机分布。zyc 试图尽可能越过障碍向里走，即，使自己的 y 坐标尽可能大。

设 zyc 能走到最内部的位置为 (x_m, y_m) ，求出 y_m^k 的期望。

答案对 P 取模。

3.2 Input

多组数据。

第一行三个正整数 T, P, K ，表示数据组数，模数和题面描述中给的参数。以后的 T 行，每行一个正整数 n ，表示走廊长度。

3.3 Output

输出 T 行，按顺序表示每一组数据的答案。

3.4 Sample Input

```
5 998244353 2
2
10
1000000
1000000000
1000000000000000000
```

3.5 Sample Output

```
207967576
475432852
721627042
```

148355385
242681415

3.6 Sample Explanation

取 $n = 2$ 为例：
以下用一个长为 n 宽为 3 的字符矩阵代表走廊。‘.’代表空地，不同数字代表不同大小的障碍。

.. 22 11 1.
.1 22 1. .1
.1 .1 11 11

以上四种可能的障碍排布，其 y_m 分别为：2，1，0，1。
总共有 72 种可能的排布，其中有 12 种 $y_m = 0$ ，21 种 $y_m = 1$ ，39 种 $y_m = 2$ 。

测试点	n≤	k	T	P	
1	10	1	5	≤1073741824	
2	1E6		10	998244353	
3	1E9				998244353
4					
5					
6					
7	1E18		50	≤1073741824	
8			200		
9	1E6	2	5		998244353
10	1E18		10		
11			50		
12			200		
13	1E9	≤5	50	998244353	
14	1E18		200	998244353	
15					
16					
17	1E9	≤20	50	≤1073741824	
18	1E18		200		
19					
20					

对于所有数据， $n \leq 1E18$ ， $K \leq 10$ ， $T \leq 200$ ， $P \leq 1073741824$ 且 P 为质数。

4 蒟蒻座蒟蒻星 (konjac)

4.1 Description

蒟蒻座蒟蒻星蒟蒻国有 n 座蒟蒻城市，标号为 $1, \dots, n$ 。 n 条双向道路连接这些城市，每条道路都有其通行时间 t_i 。这些城市和道路形成了一张无向联通图。
定义一条路径为一些城市的序列 c_1, c_2, \dots, c_L ，其中对于所有 $1 \leq i < L$ ，城市 c_i 与城市 c_{i+1} 有道路相连。

蒟蒻星的地质条件是不稳定的。频发的地质灾害（下称为“事件”）会摧毁部分道路，而蒟蒻常需要集合以统筹资源安排，计划修复道路。

具体地说，总共有 Q 次事件：

- 两条道路 $\langle p_x, q_x, t_x \rangle, \langle p_y, q_y, t_y \rangle$ 被摧毁。本题保证：摧毁这两条道路后，对于任意两座城市 i, j ，它们要么无法互达，要么恰好存在一条从 i 到 j 的路径。（即：摧毁这两条道路后，整个图被切割成若干树形联通块）
- 蒟蒻选定数量尽量少的一组地点，称其为集合地点，使得任意一个城市都能到达这一些集合地点之一。集合地点既可以选在城市中，也可以选在道路上的某个位置。
- 所有城市派出一名蒟蒻（于是总共派出了 n 名蒟蒻）前往距离其最近的集合地点，所需时间为它经过的所有道路的通行时间之和（如果只经过一条道路的一部分，就只计算这一部分的通行时间）。某集合地点的集合时间为所有前往该地点的蒟蒻出行所需时间的最大值。总集合时间为所有集合地点的集合时间之和。蒟蒻们当然希望总集合时间越短越好。
- 最后，蒟蒻们在下一次事件前迅速把两条道路修复好。

对于每次事件，你需要求出数量尽量少的一组集合地点并在数量尽量少的情况下使总集合时间最短。

4.2 Input

第一行一个正整数 n ，表示城市个数和道路条数。

接下来的 n 行，每一行三个正整数 p_i, q_i, t_i ，表示有一条道路连接城市 p_i 和 q_i ，通行时间为 t_i 。保证无重边，无自环。

接下来一行一个正整数 Q ，表示事件次数。

接下来的 Q 行，每行四个正整数 p_x, q_x, p_y, q_y ，表示连接 p_x, q_x 的道路和连接 p_y, q_y 的道路被摧毁，保证存在连接 p_x, q_x 的道路和连接 p_y, q_y 的道路且这两条道路不同。保证切除它们后，整个图被分割成若干个树形连通块。

4.3 Output

输出 Q 行，每行一个小数代表对应事件下的最短总集合时间。四舍五入到小数点后 1 位。

4.4 Sample Input

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

4 1 3 1

4.5 Sample Output

15.0

10.0

12.5

测试点	$n, Q \leq$	图的形态	特殊性质	
1	5			
2	30			
3	200			
4	500			
5	1000			
6	3000			
7				X
8				Y
9	200000	X	A	
10				
11		Y	B	
12				
13	50000		A	
14	200000			
15	50000		B	
16	200000			
17	30000			
18	60000			
19	100000			
20	300000			

对于所有数据， $3 \leq n \leq 300000$ ， $1 \leq Q \leq 300000$ ， $1 \leq p_i, q_i \leq n, 1 \leq t_i \leq 100000, 1 \leq p_x, q_x, p_y, q_y \leq n$ 。

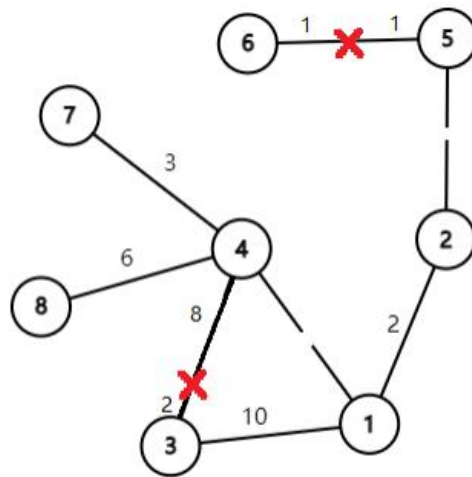
说明：

- 图的形态中的 X 型：图中环的长度不超过 5。Y 型：图中环的长度不低于 $n-100$ 。

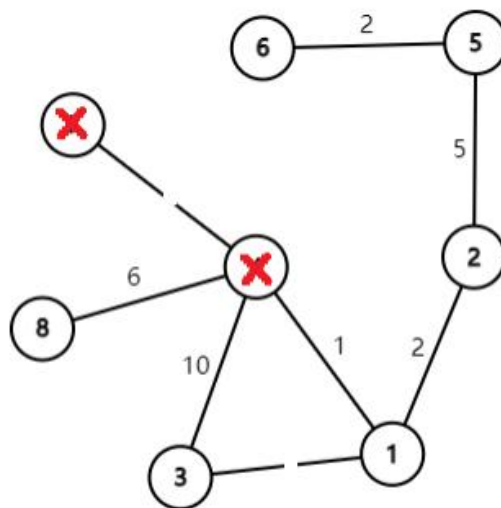
- 特殊性质 **A** : 每次摧毁的都是环边。特殊性质 **B** : 每次摧毁的恰有一条是环边。

4.6 Sample Explanation

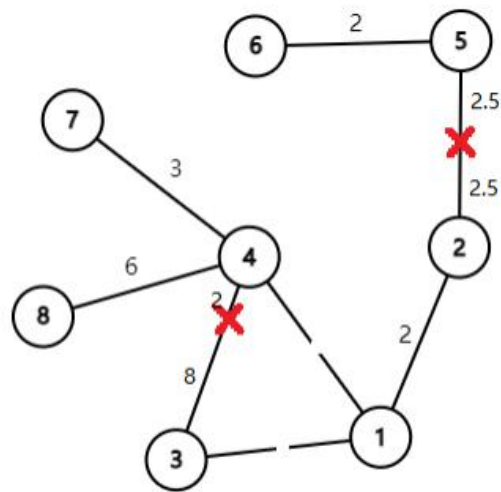
注意集合地点可能在道路中间的某个位置。



第一次事件，选择的位置是 $\langle 5,6 \rangle$ 中点和 $\langle 3,4 \rangle$ 的 $1:4$ 五等分点；



第二次事件，选择 4 和 7；



第三次事件，选择 $\langle 2, 5 \rangle$ 中点和 $\langle 3, 4 \rangle$ 的 4:1 五等分点。