

NOIP2018 模拟赛@20181004

猫

A. 龙珠 (dragon, 1s, 512MB)

板板希望能获得一个魔法值为 $\frac{A}{B}$ 的龙珠。

板板可以让两个龙珠融合，对于两个魔法值为 A 和 B 的龙珠，有以下两种融合方式：

- 暴力融合两个龙珠，得到一个魔法值为 $A + B$ 的新龙珠
- 用念力大法融合两个龙珠，得到一个魔法值为 $\frac{1}{\frac{1}{A} + \frac{1}{B}}$ 的龙珠

同时为了保证质量，每次融合两个龙珠的时候，其中必须有一个魔法值为 1 的新龙珠。

购买一个新的龙珠需要 T 元。板板希望花费尽量少的钱获得一个魔法值为 $\frac{A}{B}$ 的龙珠。

输入格式

一行 A, B, T 。

输出格式

一行答案，若无解输出 -1 。

样例输入

```
1 | 5 1 6
```

样例输出

```
1 | 30
```

子任务

对于 30% 的数据， $A, B \leq 20$ ；

对于所有数据， $A \geq 0, B > 0, A, B \leq 10^{18}, T \leq 100$ ，保证答案不超过 4×10^{18} 。

B. 跳棋 (rabbit, 1s, 512MB)

在棋盘上有三个跳棋，刚开始它们分别在 a, b, c 这三个不同的点。

每次，一个跳棋可以跳过另一个跳棋，到达轴对称的位置上，即如果 a 处的跳棋跳过 b 处的跳棋，就会到达 $2b - a$ 处。但是，一个跳棋一次只能跳过一个跳棋而不能跳过两个，并且目标位置也不能有别的跳棋。

求这三个跳棋恰好跳 k 步跳到 d, e, f 这三个位置（不要求一一对应）的方案数。

输入格式

一行 a, b, c, d, e, f, k 。

输出格式

输出不同的方案数模 $10^9 + 7$ 。

样例输入

```
1 | 0 5 8 0 8 11 3
```

样例输出

```
1 | 5
```

子任务

对于 30% 的数据, $|a|, |b|, |c|, |d|, |e|, |f| \leq 50$;

对于 100% 的数据, $|a|, |b|, |c|, |d|, |e|, |f| \leq 10^{18}, k \leq 100$ 。

C. 快递公司 (transport, 5s, 512MB)

TopTree 快递公司是一家新兴的快递公司, 这家公司以派送快速、网点众多而大受欢迎。现在, TopTree 公司的 F 省分部刚刚开业, 你——F 省分部的总裁, 正准备决定一系列事务。

F 省地处沿海, 所有城市都是分布在海岸线附近, 因此它的形状十分狭长。 n 座城市从东到西排列在海岸线周围, 按从东到西的顺序从 1 到 n 编号。为了成本的考虑, 在每座城市只能设立不超过 p 个派送地点。 TopTree 公司在第 i 座城市设立了 c_i 个派送地点。其中第 i 个城市的第 j ($j \leq c_i$) 个派送地点被标号为 $j + \sum_{k=1}^{i-1} c_k$ 。举个例子, 如果 $c_1 = 3, c_2 = 2, c_3 = 4$, 那么在第 1 座城市中的派送地点的编号为 1, 2, 3, 在第 2 座城市中的派送地点的编号为 4, 5, 在第 3 座城市中的派送地点编号为 6, 7, 8, 9。

TopTree 公司使用专用货车在派送地点之间运输货物。为了强化治安, 在 F 省中, 货车在一次运输中, 起点和终点的城市的距离不能超过 k , 即若起点为 s 、终点为 t , 则货车能从城市 s 的某个派送地点一次运输到城市 t 的某个派送地点仅当 $|s - t| \leq k$ 。因此你制定了 m 条双向运输路线, 第 i 条路线为 x_i, y_i, w_i , 表示从第 x_i 个派送地点可以通过一次货车运输花费 w_i 的时间到达第 y_i 个派送地点, 第 y_i 个派送地点也能通过一次货车运输花费 w_i 的时间到达第 x_i 个派送地点。这些路线必然符合 F 省的规定, 并且两个派送地点之间有且仅有一条路线。

现在你收到了 q 份快递需求, 第 i 份是从第 s_i 个派送地点出发, 要求送到第 t_i 个派送地点。一次派送可以经过多个城市, 也可以经过多次的货车运输。而你的任务, 就是对于每一份快递需求, 找到一种合理的运输方式, 使得所花费的时间最小。

输入格式

输入文件的第一行包含 5 个整数 n, m, k, p, q , 表示城市的个数、运输路线的个数、F 省的一次运输的距离限制、每个城市最多设立的派送地点数和快递需求的个数;

第二行包含 n 个整数 c_1, c_2, \dots, c_n , 表示每个城市的派送地点的个数;

接下来 m 行，每行 3 个整数 x_i, y_i, w_i ，表示一条在派送地点 x_i 和派送地点 y_i 之间的、花费时间为 w_i 的运输路线，保证每条运输路线符合 F 省的规定，并且两个派送地点之间有且仅有一条路线；

接下来 q 行，每行 2 个整数 s_i, t_i ，表示一条从派送地点 s_i 到派送地点 t_i 的快递需求。

输出格式

输出文件包含 q 行，第 i 行为第 i 条快递需求的最小花费时间，如果不存在一个运输的方案，则输出一行“-1”（不包含引号）。

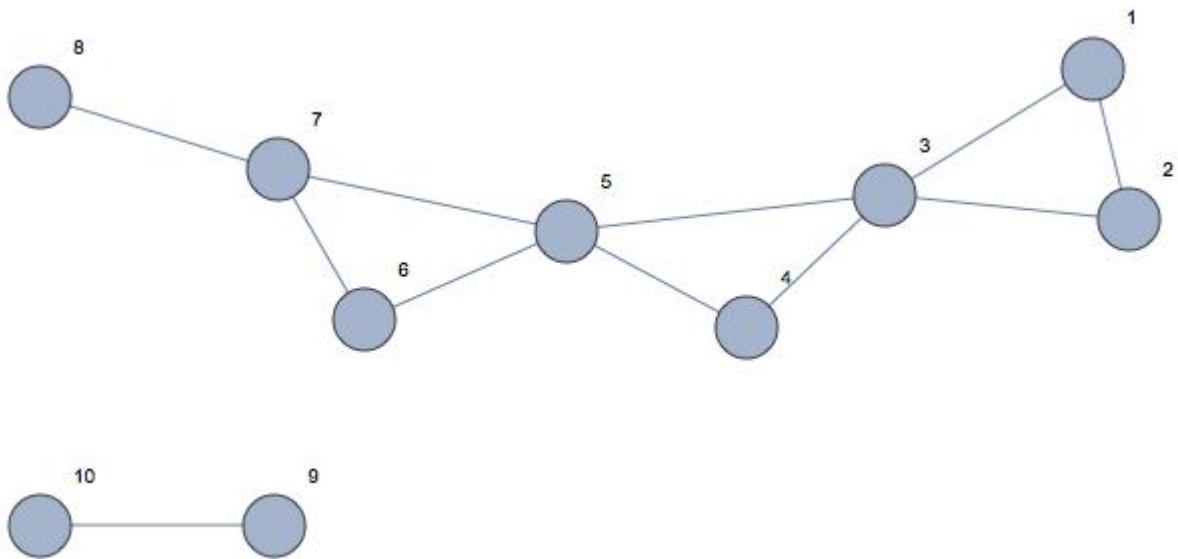
样例输入

```
1 | 5 11 1 2 5
2 | 2 2 2 2 2
3 | 1 2 1
4 | 1 3 2
5 | 2 3 1
6 | 3 4 1
7 | 3 5 2
8 | 4 5 1
9 | 5 6 1
10 | 5 7 2
11 | 6 7 1
12 | 7 8 1
13 | 9 10 1
14 | 1 4
15 | 2 8
16 | 5 7
17 | 9 4
18 | 3 2
```

样例输出

```
1 | 3
2 | 6
3 | 2
4 | -1
5 | 1
```

样例说明



上图是派送地点之间的图，花费的时间未画出。下面的解释中， $a \rightarrow b$ 表示 a 通过一次货车运输到达 b 。

对于第 1 个派送请求，一种可行的最优方案为 $1 \rightarrow 3 \rightarrow 4$ ，花费时间 $2 + 1 = 3$ ；

对于第 2 个派送请求，一种可行的最优方案为 $2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 7 \rightarrow 8$ ，花费时间 $1 + 1 + 1 + 2 + 1 = 6$ ；

对于第 3 个派送请求，一种可行的最优方案为 $5 \rightarrow 6 \rightarrow 7$ ，花费时间 $1 + 1 = 2$ ；

对于第 4 个派送请求，没有任何可行方案；

对于第 5 个派送请求，唯一的最优方案为 $3 \rightarrow 2$ ，花费时间 1。

子任务

测试点编号	n	m	k	p	q	w	备注
1	≤ 3	≤ 8	$= 1$	≤ 3	≤ 5	≤ 5000	无
2	≤ 100	≤ 3000	≤ 3	≤ 5	≤ 200		
3					≤ 2000		
4					≤ 200000		
5							
6					≤ 1000		≤ 10000
7	≤ 30000	≤ 200000	≤ 5	≤ 200000	≤ 5000		所有 $s_i = 1$
8							
9							
10	≤ 100000		$= 1$				无
11	≤ 30000	≤ 100000	$= 0$	≤ 5	≤ 200000		
12							
13							
14	≤ 100000	≤ 200000	$= 1$				
15	≤ 30000	≤ 100000	$= 1$	$= 2$	≤ 200000		所有 $c_i = p$
16				$= 3$			
17							
18							
19							
20		≤ 200000					
21					≤ 100000		无
22	≤ 10000	≤ 50000	≤ 3	≤ 5			
23							
24	≤ 30000	≤ 200000					
25							
						≤ 200000	