

全国青少年信息学奥林匹克联赛模拟赛

mNOIP - 20171008

提高组 第二试

fstqwq

题目名称	入阵曲	将军令	星空
题目类型	传统型	传统型	传统型
可执行文件名	rally	general	starlit
输入文件名	rally.in	general.in	starlit.in
输出文件名	rally.out	general.out	starlit.out
单点时限	1.0 秒	1.0 秒	1.0 秒
内存限制	256MB	256MB	256MB
测试点数目	20	20	25
单点分值	5	5	4

提交源程序文件名

对于 C++ 语言	rally.cpp	general.cpp	starlit.cpp
对于 C 语言	rally.c	general.c	starlit.c
对于 Pascal 语言	rally.pas	general.pas	starlit.pas

编译选项

对于 C++ 语言	-O2 -lm	-O2 -lm	-O2 -lm
对于 C 语言	-O2 -lm	-O2 -lm	-O2 -lm
对于 Pascal 语言	-O2	-O2	-O2

注意事项:

1. 文件名（程序名和输入输出文件名）必须使用英文小写。
2. 除非特殊说明，结果比较方式均为忽略行末空格及文末回车的全文比较。
3. C/C++中函数 main()的返回值必须是 int，程序正常结束时的返回值必须是 0。
4. 评测时参考机器配置为：CPU Intel i5-3470 @ 3.20 GHz，内存 4G，上述时限以此配置为准。
5. 只提供 Linux 格式的附加样例文件。
6. 如果无特殊声明，评测将在 Linux 下进行。
7. 如果无特殊声明，所有输入数据中出现的整数均为正整数。

入阵曲(rally)

【题目描述】

丹青千秋酿，一醉解愁肠。
无悔少年枉，只愿壮志狂。

小 F 很喜欢数学，但是到了高中以后数学总是考不好。

有一天，他在数学课上发起了呆；他想起了过去的一年。一年前，当他初识算法竞赛的时候，觉得整个世界都焕然一新。这世界上怎么会有这么多奇妙的东西？曾经自己觉得难以解决的问题，被一个又一个算法轻松解决。

小 F 当时暗自觉得，与自己的幼稚相比起来，还有好多要学习的呢。

一年过去了，想想都还有点恍惚。

他至今还能记得，某天晚上听着入阵曲，激动地睡不着觉，写题写到鸡鸣时分都兴奋不已。也许，这就是热血吧。

1570479	FirstLast	4034	Accepted	16196 kb	2512 ms	C++/Edit	3186 B	2016-08-01 03:08:46
---------	-----------	------	----------	----------	---------	----------	--------	---------------------

也就是在那个时候，小 F 学会了矩阵乘法。让两个矩阵乘几次就能算出斐波那契数列的第 10^{100} 项，真是奇妙无比呢。

不过，小 F 现在可不想手算矩阵乘法——他觉得好麻烦。取而代之的，是一个简单的小问题。他写写画画，画出了一个 $n \times m$ 的矩阵，每个格子里都有一个不超过 k 的正整数。小 F 想问问你，这个矩阵里有多少个不同的子矩形中的数字之和是 k 的倍数？

如果把一个子矩形用它的左上角和右下角描述为 (x_1, y_1, x_2, y_2) ，其中 $x_1 \leq x_2, y_1 \leq y_2$ ；那么，我们认为两个子矩形是不同的，当且仅当他们以 (x_1, y_1, x_2, y_2) 表示时不同；也就是说，只要两个矩形以 (x_1, y_1, x_2, y_2) 表示时相同，就认为这两个矩形是同一个矩形，你应该在你的答案里只算一次。

【输入格式】

从文件 *rally.in* 中读入数据。

输入第一行，包含三个正整数 n, m, k 。

输入接下来 n 行，每行包含 m 个正整数，第 i 行第 j 列表示矩阵中第 i 行第 j 列中所填的正整数 $a_{i,j}$ 。

【输出格式】

输出到文件 *rally.out* 中。

输入一行一个非负整数，表示你的答案。

【样例 1 输入】

```
2 3 2
1 2 1
2 1 2
```

【样例 1 输出】

6

【样例 1 说明】

这些矩形是符合要求的：

(1, 1, 1, 3), (1, 1, 2, 2), (1, 2, 1, 2), (1, 2, 2, 3), (2, 1, 2, 1), (2, 3, 2, 3)。

【样例 2】

见选手目录下的 *rally/rally2.in* 与 *rally/rally2.ans*。

【数据范围与约定】

子任务会给出部分测试数据的特点。如果你在解决题目中遇到了困难，可以尝试只解决一部分测试数据。

每个测试点的数据规模及特点如下表：

测试点	n	m	k	特殊性质	
1	2	2	$\leq 10^6$	\times	
2	4	2			
3		4			
4	≤ 16	2	≤ 2	\checkmark	
5		≤ 16	$\leq 10^6$		
6					
7	≤ 80	2	≤ 2	\times	
8		≤ 80	$\leq 10^6$		
9			≤ 2	\checkmark	
10			$\leq 10^6$		
11	≤ 400	2	≤ 2	\times	
12		≤ 400			
13		≤ 2	\checkmark		
14		$\leq 10^6$			
15		$\leq 10^6$			
16		≤ 400	≤ 400	\times	
17			≤ 400		
18			$\leq 10^6$		
19			$\leq 10^6$		
20					

特殊性质：保证所有 $a_{i,j}$ 均相同。

将军令(general)

【题目描述】

历史/落在/赢家/之手
至少/我们/拥有/传说
谁说/败者/无法/不朽
拳头/只能/让人/低头
念头/却能/让人/抬头
抬头/去看/去爱/去追
你心中的梦

又想起了四月。

如果不是省选，大家大概不会这么轻易地分道扬镳吧？

只见一个又一个昔日的队友离开了机房。

凭君莫话封侯事，一将功成万骨枯。

梦里，小 F 成了一个给将军送密信的信使。

现在，有两封关乎国家生死的密信需要送到前线大将军帐下，路途凶险，时间紧迫。小 F 不因为自己的祸福而避趋之，勇敢地承担了这个任务。

不过，小 F 实在是太粗心了，他一不小心把两封密信中的一封给弄掉了。

小 F 偷偷打开了剩下的那封密信。他发现一副十分详细的地图，以及几句批文——原来是战场周围的情报地图。他仔细看后发现，在这张地图上标记了 n 个从 1 到 n 标号的驿站， $n-1$ 条长度为 1 里的小道，每条小道双向连接两个不同的驿站，并且驿站之间可以通过小道两两可达。

小 F 仔细辨认着上面的批注，突然明白了丢失的信的内容了。原来，每个驿站都可以驻扎一个小队，每个小队可以控制距离不超过 k 里的驿站。如果有驿站没被控制，就容易产生危险——因此这种情况应该完全避免。而那封丢失的密信里，就装着朝廷数学重臣留下的精妙的排布方案，也就是用了最少的小队来控制所有驿站。

小 F 知道，如果能计算出最优方案的话，也许他就能将功赎过，免于死罪。他找到了你，你能帮帮他吗？

当然，小 F 在等待你的支援的过程中，也许已经从图上观察出了一些可能会比较有用的性质，他会通过一种特殊的方式告诉你。

【输入格式】

从文件 *general.in* 中读入数据。

输入第 1 行一个正整数 n, k, t ，代表驿站数，一支小队能够控制的最远距离，以及特殊性质所代表的编号。关于特殊性质请参照数据范围。

输入第 2 行至第 n 行，每行两个正整数 u_i, v_i ，表示在 u_i 和 v_i 间，有一条长度为一里的小道。

【输出格式】

输出到文件 *general.out* 中。

输出一行，为最优方案下需要的小队数。

【样例 1 输入】

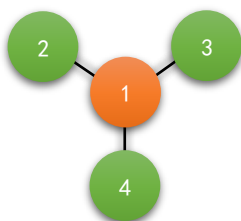
```
4 1 0
1 2
1 3
1 4
```

【样例 1 输出】

```
1
```

【样例 1 说明】

如图。由于一号节点到周围的点距离均是 1，因此可以控制所有驿站。



【样例 2 输入】

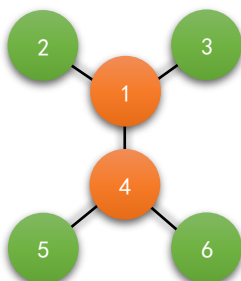
```
6 1 0
1 2
1 3
1 4
4 5
4 6
```

【样例 2 输出】

```
2
```

【样例 2 说明】

如图，和样例 1 类似。



【数据范围】

子任务会给出部分测试数据的特点。如果你在解决题目中遇到了困难，可以尝试只解决一部分测试数据。

关于 t 的含义如下：

$t = 0$ ：该测试点没有额外的特殊性质；

$t = 1$ ：保证最多 8 个点的所连接的小道超过 1 条；

$t = 2$ ：保证所有点到 1 号点的距离不超过 2。

每个测试点的数据规模及特点如下表：

测试点	n	k	t
1	$\leq 10^5$	0	0
2	≤ 16	1	
3			
4	≤ 100		
5			
6	$\leq 10^5$		
7			
8			
9			
10	≤ 16		2
11			
12	≤ 100		
13			
14	$\leq 10^5$	1	
15		0	
16	≤ 16		3
17	≤ 100		
18			
19	$\leq 10^5$	≤ 20	1
20			0

星空(starlit)

【题目描述】

命运偷走如果只留下结果，
时间偷走初衷只留下了苦衷。
你来过，然后你走后，只留下星空。

逃不掉的那一天还是来了，小 F 看着夜空发呆。
天上空荡荡的，没有一颗星星——大概是因为天上吹不散的乌云吧。
心里吹不散的乌云，就让它在那里吧，反正也没有机会去改变什么了。

小 C 拿来了一长串星型小灯泡，假装是星星，递给小 F，想让小 F 开心一点。不过，有着强迫症的小 F 发现，这串一共 n 个灯泡的灯泡串上有 k 个灯泡没有被点亮。小 F 决定和小 C 一起把这个灯泡串全部点亮。

不过，也许是因为过于笨拙，小 F 只能将其中连续一段的灯泡状态给翻转——点亮暗灯泡，熄灭亮灯泡。经过摸索，小 F 发现他一共能够翻转 m 种长度的灯泡段中灯泡的状态。

小 C 和小 F 最终花了很长很长很长很长很长的时间把所有灯泡给全部点亮了。他们想知道他们是不是蠢了，因此他们找到了你，让你帮忙算算：在最优的情况下，至少需要几次操作才能把整个灯泡串给点亮？

【输入格式】

从文件 *starlit.in* 中读入数据。
输入第 1 行三个正整数 n, k, m 。
输入第 2 行 k 个正整数，第 i 个数表示第 i 个被没点亮的灯泡的位置 a_i 。
输入第 3 行 m 个正整数，第 i 个数表示第 i 种操作的长度 b_i 。
保证所有 b_i 互不相同；保证对于 $1 \leq i < k$ ，有 $a_i < a_{i+1}$ ；保证输入数据有解。

【输出格式】

输出到文件 *starlit.out* 中。
输出一行一个非负整数，表示最少操作次数。

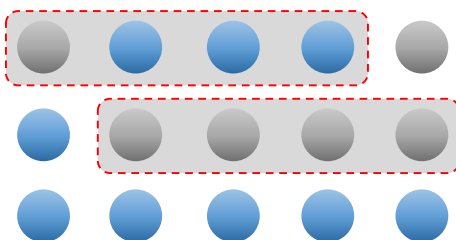
【样例 1 输入】

```
5 2 2
1 5
3 4
```

【样例 1 输出】

```
2
```

【样例 1 解释】



【样例 2】

见选手目录下的 *starlit/starlit2.in* 与 *starlit/starlit2.ans*。

【数据范围】

子任务会给出部分测试数据的特点。如果你在解决题目中遇到了困难，可以尝试只解决一部分测试数据。

每个测试点的数据规模及特点如下表：

测试点	n	m	b_i	k	特殊性质		
1	2	2	≤ 2	≤ 2	√		
2	4	1	≤ 4	≤ 4	×		
3		≤ 4					
4	16	1	≤ 16	≤ 8			
5		≤ 16					
6							
7	200	1	≤ 2			√	
8			≤ 200				
9		≤ 4	≤ 4				
10							
11							
12		≤ 64	≤ 200		×		
13							
14	40000	1	≤ 2				√
15			≤ 40000				×
16							
17		≤ 4	≤ 4				
18							
19							
20		≤ 64	≤ 40000			√	
21							
22							
23							
24							
25		≤ 64	≤ 40000		×		

特殊性质：保证答案小于 4。