

CSP 入门组模拟赛

试题名称	手机	方方游戏	传球游戏	找位置
源程序名	iphone.cpp	square.cpp	ball.cpp	relocation.cpp
输入文件名	iphone.in	square.in	ball.in	relocation.in
输出文件名	iphone.out	square.out	ball.out	relocation.out
时限	1S	1S	1S	1S
空间限制	256M	256M	256M	256M
测试点个数	10	10	10	10
满分	100	100	100	100

1、手机 (iphone)

问题描述

小 x 有学霸姐姐帮忙复习功课，终于在一次考试中取得了还算理想的成绩。他的家长为了奖励他，帮他买了一部手机。考虑到小 x 天然呆的程度，自然不能买带虚拟键盘的 iphone，所以，他的手机是有实体键盘的老式手机。

按键对应如下：

2-ABC

3-DEF

4-GHI

5-JKL

6-MNO

7-PQRS

8-TUV

9-WXYZ

0-空格

连续按某个键 k 次 (k 小于等于该键所能代表的总字母数)，就能输入该键所表示的第 k 个字母。例如按键66777则能输入 NR。

现在给出需要输入的字符串 (所有的字母均为大写字母)，请给出按键的顺序。

输入

输入文件一行一个字符串，表示需要输入手机的字符串。

输出

输出文件一行一个字符串，表示需要按的手机键的序列。

样例输入

LUCKYSTAR

样例输出

5558822255999777782777

数据范围

对于100%的数据，保证输入文件小于1M。

2、方方游戏 (square)

问题描述

众所周知，方方热爱各种游戏，特别擅长于俄罗斯方块，小 x 每次都只能甘拜下风。

为了难为一下方方，小 x 设计了一种新的俄罗斯方块类游戏，姑且称为“方方游戏”。关键规则几乎都与俄罗斯方块一样，唯一的不同是下落物体的形状是不规则的。

方方对这个新的游戏有点丈二和尚摸不着头脑，因此求助于你。

这里我们将问题简化，给出某个下落物体落下来之前游戏区内剩余的方块情况，以及即将落下来的物体的形状，并给定下落的位置，不能调整下落物体的方向和下落位置，请输出该物体下落稳定后第一次消除的行数，即不考虑第一次消除后，下落物体剩余部分由于失去连接导致下落而继续消除的行数，假定下落物体一开始在无穷高处。

输入

输入文件的第一行为3个整数 w 、 n 、 m ，表示游戏区的宽度，游戏区内剩余的方块个数，下落物体的方块个数。

接下来的 n 行，每行两个整数 x 和 y ，表示第 x 行第 y 列有一个方块。注意，列数从左到右递增，行数从下往上递增，数据保证 n 个 (x, y) 两两不同。

接下来一行一个整数，表示下落物体中某个参照方块所在的列数。

再接下来的 $m-1$ 行，每行两个整数，分别表示其他方块与该参照方块的行之差和列之差，均由参照方块的行数和列数做减数。

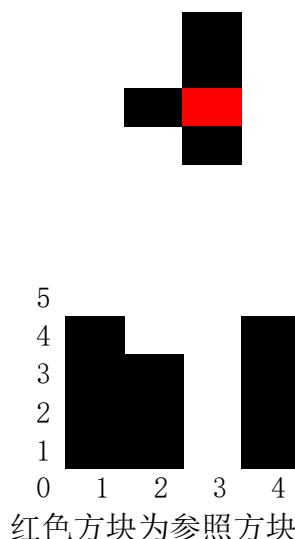
以上所有的整数均小于等于100。

输出

输出一行一个数，表示第一次消除的行数。输入数据保证原本剩余的方块不可消除。

样例输入

```
4 11 5
1 1
1 2
1 4
2 1
2 2
2 4
3 1
3 2
3 4
```



4 1
4 4
3
0 -1
-1 0
1 0
2 0

样例输出

2

数据范围

对于70%的数据满足： $1 \leq w \leq 10, 1 \leq n \leq 30, 1 \leq m \leq 30$;

对于100%的数据满足： $1 \leq w \leq 20, 0 \leq n \leq 100, 1 \leq m \leq 100$ 。

3、传球游戏 (ball)

题目描述

上体育课的时候，小 x 的老师经常带着同学们一起做游戏。这次，老师带着同学们一起做传球游戏。

游戏规则是这样的：n 个同学站成一个圆圈，其中的一个同学手里拿着一个球，当老师吹哨子时开始传球，每个同学可以把球传给自己左右的两个同学中的一个（左右任意），当老师再次吹哨子时，传球停止，此时，拿着球没传出去的那个同学就是败者，要给大家表演一个节目。

聪明的小 x 提出一个有趣的问题：有多少种不同的传球方法可以使得从小 x 手里开始传的球，传了 m 次以后，又回到小 x 手里。两种传球的方法被视作不同的方法，当且仅当这两种方法中，接到球的同学按接球顺序组成的序列是不同的。比如有 3 个同学 1 号、2 号、3 号，并假设小 x 为 1 号，球传了 3 次回到小 x 手里的方式有 1→2→3→1 和 1→3→2→1，共 2 种。

输入

输入文件一行两个整数 n 和 m，用一个空格隔开的。

输出

输出文件一行一个整数，表示符合题意的方法数。

样例输入

3 3

样例输出

2

数据范围

对于 100% 的数据满足： $3 \leq n \leq 30$ ， $1 \leq m \leq 30$ 。

4、找位置 (relocation)

【问题描述】

FJ 想找一个最好的位置来建新农场，这样他每天可以少走些路。FJ 所在的区域，有 N 个城镇，城镇之间有 M 条双向路相连，所有城镇都可以借助一些路相互连接。

FJ 需要你的帮助来选择最合适建新农场的城镇。 K 个城镇中有超市，FJ 每天都会去这些超市。他计划每天从新农场出发，访问包含超市的 K 个城镇，然后返回新农场。FJ 可以按照任意的顺序访问这些超市。FJ 只会在 $N-K$ 个城镇中，选择一个城镇来建新农场。因为其他城镇的房价，比较低一些。如果他把农场建在最优的位置，而且尽可能明智的选择行走路线。

请你编程帮 FJ 计算，他每天需要行走的路线长度。

【输入】

第1行：三个空格隔开的整数， N ， M 和 K 。

第2..1+ K 行：第 $i+1$ 行包含一个整数，范围 $1..N$ ，表示包含第 i 个超市的城镇。每个超市在不同城镇。

第2+ K ..1+ K + M 行：每行包含三个整数 i ， j 和 L ($1 \leq i, j \leq N$ ， $1 \leq L \leq 1000$)，表示城镇 i 和城镇 j 之间存在一条长度为 L 的路，每两个整数之间用一个空格隔开。

【输出】

输出一行一个数，表示他把农场建在最优的位置，每天需要行走的最短路线长度。

【输入输出样例】

relocation.in	relocation.out
5 6 3	12
1	
2	
3	
1 2 1	
1 5 2	
3 2 3	
3 4 5	
4 2 7	
4 5 10	

【样例解释】

FJ 在5号城镇建立农场。他每天的行走路线为5-1-2-3-2-1-5，路线长度为12。

【数据规模】

对于20%的数据满足： $N \leq 50$ ， $M \leq 500$ ；

对于50%的数据满足： $N \leq 1000$ ， $M \leq 5000$ ；

对于100%的数据满足： $1 \leq N \leq 10000$ ， $1 \leq M \leq 50000$ ， $1 \leq K \leq 5$ 。