

### 第3题：多边形(polygon)，运行时限 1s，内存上限 512M，100 分。

#### 【问题描述】

小 R 与小 W 在玩游戏。

他们有一个边数为 $n$ 的凸多边形，其顶点沿逆时针方向标号依次为 $1, 2, 3, \dots, n$ 。最开始凸多边形中有 $n$ 条线段，即多边形的 $n$ 条边。这里我们用一个有序数对 $(a, b)$ （其中 $a < b$ ）来表示一条端点分别为顶点 $a, b$ 的线段。

在游戏开始之前，小 W 会进行一些操作。每次操作时，他会选中多边形的两个互异顶点，给它们之间连一条线段，并且所连的线段不会与已存的线段重合、相交（只拥有一个公共端点不算作相交）。他会不断重复这个过程，直到无法继续连线，这样得到了状态 $S_0$ 。 $S_0$ 包含的线段为凸多边形的边与小 W 连上的线段，容易发现这些线段将多边形划分为一个个三角形区域。对于其中任意一个三角形，其三个顶点为 $i, j, k (i < j < k)$ ，我们可以给这个三角形一个标号 $j$ ，这样一来每个三角形都被标上了 $2, 3, \dots, n-1$ 中的一个，且没有标号相同的两个三角形。

小 W 定义了一种“旋转”操作：对于当前状态，选定4个顶点 $a, b, c, d$ ，使其满足 $1 \leq a < b < c < d \leq n$ 且它们两两之间共有5条线段—— $(a, b), (b, c), (c, d), (a, d), (a, c)$ ，然后删去线段 $(a, c)$ ，并连上线段 $(b, d)$ 。那么用有序数对 $(a, c)$ 即可唯一表示该次“旋转”。我们称这次旋转为 $(a, c)$ “旋转”。显然每次进行完“旋转”操作后多边形中依然不存在相交的线段。

当小 W 将一个状态作为游戏初始状态展示给小 R 后，游戏开始。游戏过程中，小 R 每次可以对当前的状态进行“旋转”。在进行有限次“旋转”之后，小 R 一定会得到一个状态，此时无法继续进行“旋转”操作，游戏结束。那么将每一次“旋转”所对应的有序数对按操作顺序写下，得到的序列即为该轮游戏的操作方案。

为了加大难度，小 W 以 $S_0$ 为基础，产生了 $m$ 个新状态。其中第 $i$ 个状态 $S_i$ 为对 $S_0$ 进行一次“旋转”操作后得到的状态。你需要帮助小 R 求出分别以 $S_0, S_1, \dots, S_m$ 作为游戏初始状态时，小 R 完成游戏所用的最少“旋转”次数，并根据小 W 的心情，有时还需求出“旋转”次数最少的不同操作方案数。由于方案数可能很大，输出时请对 $1,000,000,007$ 取模。

#### 【程序文件名】

源程序文件名为 polygon.cpp/c/pas。

#### 【输入格式】

输入文件名为 polygon.in。

第一行一个整数 $W$ ，表示小 W 的心情。若 $W$ 为0则只需求出最少的“旋转”次数，若 $W$ 为1则还需求出“旋转”次数最少时的不同操作方案数。

第二行一个正整数 $n$ ，表示凸多边形的边数。

接下来 $n-3$ 行，每行两个正整数 $x, y$ ，表示小 W 在 $S_0$ 中连的一条线段，端点分别为 $x, y$ 。保证该线段不与已存的线段重合或相交。

接下来一行一个整数 $m$ ，表示小 W 以 $S_0$ 为基础产生的新状态个数。

接下来 $m$ 行，每行两个整数。假设其中第 $i$ 行为 $a, b$ ，表示对 $S_0$ 进行 $(a, b)$ “旋转”后得到 $S_i$ 。

#### 【输出格式】

输出文件名为 polygon.out。

输出共 $m+1$ 行。

若 $W$ 为0则每一行输出一个整数，第 $i(i = 1, 2, \dots, m, m + 1)$ 行输出的整数表示 $S_{i-1}$ 作为初始局面的最少“旋转”次数。

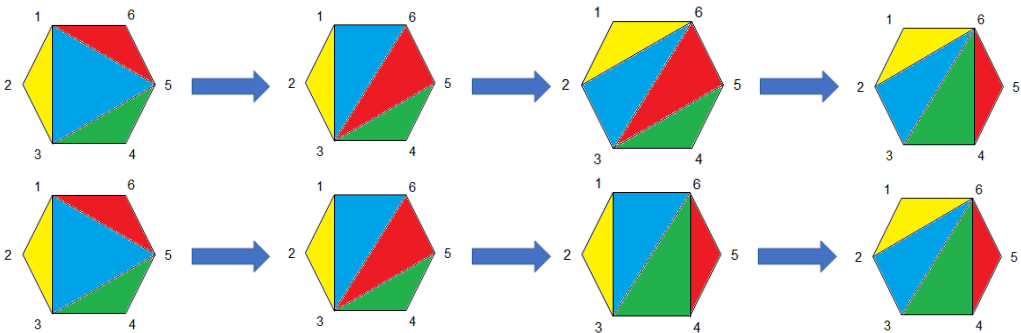
若 $W$ 为1则每一行输出两个整数，第 $i(i = 1, 2, \dots, m, m + 1)$ 行输出的两个整数依次表示 $S_{i-1}$ 作为初始局面的最少“旋转”次数、“旋转”次数最少的不同操作方案数对1,000,000,007取模的结果。

【输入输出样例1】

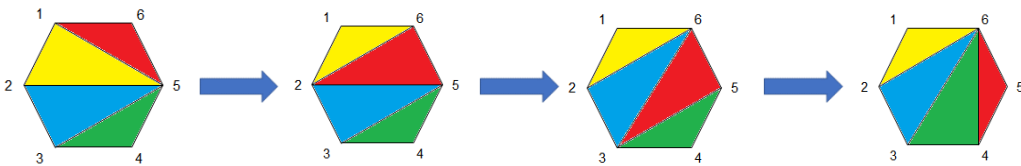
polygon.in	polygon.out
1	3 2
6	3 1
1 3	
1 5	
3 5	
1	
1 3	

【样例解释1】

以 $S_0$ 为初始状态，最少“旋转”次数为3，有2种方案，如下：



以 $S_1$ 为初始状态，最少“旋转”次数为3，有1种方案，如下：



以上图中黄色、蓝色、绿色、红色三角形对应标号分别为2,3,4,5。

【输入输出样例2】

polygon.in	polygon.out
1	8 210
12	7 210
1 10	8 70
1 6	8 105
1 3	8 140
3 6	
3 5	
6 10	
6 8	
8 10	

10 12	
4	
1 10	
1 3	
6 8	
1 6	

### 【数据范围】

测试点编号 $id$	$W$	$n$	$m$	测试点编号 $id$	$W$	$n$	$m$
1	1	$= id + 8$	$= 0$	11	0	$\leq 10000$	$\leq 100000$
2				12			
3			$\leq n$	13	1		$\leq 1000$
4				14			
5		15		$\leq 100000$		$= 0$	
6		16				$\leq 100000$	
7		17					
8		18					
9		$= 0$	19				
10			20				

对于所有输入数据，保证： $3 \leq n \leq 100000, 0 \leq m \leq 100000$ 。

### 【编译命令】

对于c++语言： `g++ -o polygon polygon.cpp -lm`

对于c语言： `gcc -o polygon polygon.c -lm`

对于pascal语言： `fpc polygon.pas`