

## Problem A. 百年校庆

Input file:           standard input  
Output file:         standard output  
Time limit:          1 second  
Memory limit:       512 megabytes

一世纪规格功夫，新百年世界一流。



在庆祝哈尔滨工业大学百年校庆之际，ACM 集训队正在准备哈工大（深圳）第二届程序设计大赛。

PBH 对我们说，在这个重大的历史转折点举办比赛，要友好一点才可以嘛。

所以，我们决定，将这道题献给哈工大百年校庆。

所以，在这一道题里，只用输出 “HIT 100th anniversary!!!!!!”（不包含引号）就可以开开心心的通过这一道题！

让我们共同庆祝哈工大百年生日快乐！

### Input

输入文件包含一个数字。

### Output

输出 “HIT 100th anniversary!!!!!!”

### Example

standard input	standard output
19202020	HIT 100th anniversary!!!!!!

### Note

没错，每人领一个气球回去才显得喜庆！

## Problem B. 闰年统计

Input file:           standard input  
Output file:         standard output  
Time limit:          1 second  
Memory limit:       512 megabytes

一世纪规格功夫，新百年世界一流。

一个世纪有多少天来着？你不禁好奇了起来。

众所周知，一个世纪有一百年，一年有 365 天。可如果这一年是闰年，那么他就有了 366 天。

为了更好的统计一个世纪有多少天，你决定写一个程序，用来判断给定的两个年份之间有多少个闰年。

### Input

输入包含一行，以空格分开的两个整数  $L, R$  ( $1582 \leq L \leq R \leq 9999$ )，表示起始年份和终止年份。

### Output

输出一行一个整数，表示从起始年份到终止年份之间，有多少个闰年（包含起始年份和终止年份）。

### Example

standard input	standard output
1871 1908	9
1920 2020	26

### Note

样例 1 中的 9 个闰年为：1872 年、1876 年、1880 年、1884 年、1888 年、1892 年、1896 年、1904 年、1908 年。注意 1900 年不是闰年。

ps: 如果你不知道闰年是什么，说明你没有好好复习 C 语言。

## Problem C. 课程展示

Input file:           standard input  
Output file:         standard output  
Time limit:          1 second  
Memory limit:       512 megabytes

为了庆祝哈工大的百年校庆，弘扬以爱国主义为核心的民族精神和以改革创新为核心的时代精神，在思修课上，老师安排了期末展示活动。

在思修课的期末展示中，一共有  $n$  位同学被选作为评委。

对于正在进行展示的小组，第  $i$  位评委会给出一个整数  $a_i$ ，作为这一组的评分。

在助教收齐了所有评委的评分后，他会去掉一个最高分，去掉一个最低分，将剩下的评分取平均数作为这个小组的最终成绩。

身为助教的小 P 觉得手算这么多加法和除法会令他十分疲惫，他决定委托你写一个程序帮帮他。

### Input

输入包含两行，第一行为一个正整数  $n(3 \leq n \leq 10^5)$ ，第二行包含  $n$  个空格分开的整数  $a_i(|a_i| \leq 10^9)$ 。

### Output

输出包含一行一个整数，表示最终得分。我们保证最终得分一定恰好是一个整数（尽管现实中很可能不是）。

### Example

standard input	standard output
5 1 2 3 4 1000	3
5 -1 -1 -1 11 11	3

### Note

对于样例 1，去掉一个最高分和一个最低分后，剩余的分数为 2 3 4 平均值为 3。

对于样例 2，去掉一个最高分和一个最低分后，剩余的分数为 -1 -1 11，平均值为 3。

对于 C 或 C++ 语言，由于在 linux 下评测，使用 scanf 或 printf 输入输出 long long 类型时请使用 %lld 而不是 %I64d。

## Problem D. 很 没 精 神

Input file: standard input  
Output file: standard output  
Time limit: 1 second  
Memory limit: 512 megabytes

由于昨晚在夜之城度过，小 L 第二天早上很没精神。为了防止错过早八课中的重要知识点，他决定将整节课录音。

课后，在听录音时，小 L 发现了一个叫“倒放”的功能，他觉得这个功能十分有趣。

在倒放时，录音中的一段音频会被反过来播放。比如，老师说“waibibabu”，那么在倒放时，就变成了“ubabibiaw”。

小 L 惊奇地发现，老师讲课中的某些部分，正放和倒放居然是一样的。比如“abbba”就具有这样的性质。他十分好奇，在整个录音中，最长的满足“正放和倒放一样”的部分有多长。由于小 L 还在夜之城，他希望你写个程序帮帮他。

ps: 我们保证录音中只有英语小写字母，没有空格、分号、逗号等。

### Input

输入包含一行字符串  $S$  ( $|S| \leq 5000$ ) 表示老师上课的录音，保证  $S$  中只有小写英文字母。

### Output

输出一行一个整数，表示录音中最长的部分的长度，满足正放、倒放相同。

### Example

standard input	standard output
guigeyange	1
ggyggfdjgctt	5
guochengtaotaigcttttcg	8

### Note

对于样例 1，满足性质的可以是任何一个长度为 1 的部分。

对于样例 2，满足性质的最长部分为 ggygg。

对于样例 3，满足性质的最长部分为 gcttttcg。

## Problem E. 工地大学

Input file: standard input  
Output file: standard output  
Time limit: 1 second  
Memory limit: 512 megabytes

哈尔滨工地大学（深圳）在旧校区旁边新开辟了一块大小为  $n \times m$  的矩形工地。在地图上，工地四个顶点的坐标分别为  $(0, 0)$ ,  $(0, m)$ ,  $(n, 0)$ ,  $(n, m)$ 。

在哈工大百年校庆之际，为了扩大招生规模，学校希望在工地上修建两栋教学楼。

在地图上，教学楼可以表示成矩形，其中一栋教学楼边长为  $a \times b$ ，另一栋为  $c \times d$ 。

在建设教学楼时，学校希望两栋教学楼的四个顶点要落在矩形工地的坐标的整数点上，且两栋教学楼的边界要与矩形工地的边界平行，并且两栋教学楼不能互相重叠（但是边、顶点可以重合；教学楼的边、顶点可以与工地的边、顶点重合）。

为了统计同学们对两栋新教学楼位置的意愿，学校需要先统计所有可能的位置数量。辅导员找到了聪明机智的你，希望你能够完成这个神圣而重大的任务。

由于答案很大，你只需要输出答案对 998244353 取模后的结果（答案除以 998244353 的余数）。ps: 整数点指的是坐标系上横坐标和纵坐标都是整数的点。

### Input

每个测试点有多组测试数据，首先第一行会输入一个数  $t$  ( $1 \leq t \leq 10^5$ )— 测试数据的数量。

对于每个数据，只有一行，包含 6 个正整数  $n, m, a, b, c, d$  ( $1 \leq n, m, a, b, c, d \leq 10^9$ )。  $n, m$  表示矩形工地的大小；  $a, b$  表示一栋教学楼的边长；  $c, d$  表示另一栋教学楼的边长。

### Output

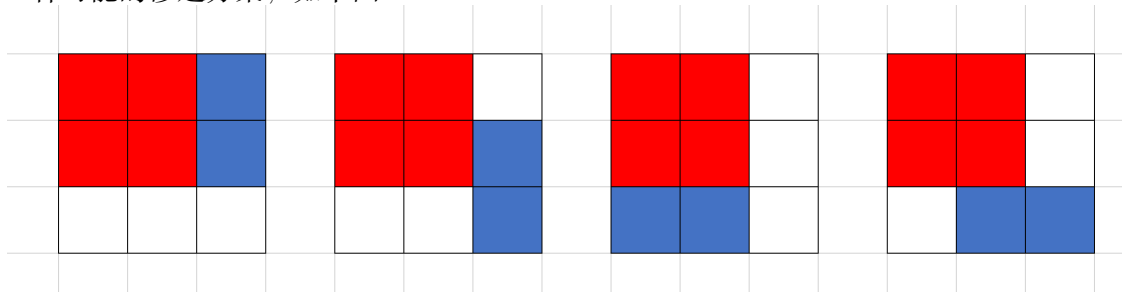
输出一共  $t$  行，对于每一组测试数据，输出一行，这一行输出一个整数为最终的答案。

### Example

standard input	standard output
2	20
3 3 2 2 1 1	16
3 3 2 2 1 2	

### Note

对于第二个测试样例：假设用红色矩形和蓝色矩形代表两个教学楼；当红色矩形在左上角的时候，一共有 4 种可能的修建方案，如下图：



红色矩形除了放在左上角，还有可以放在右上角，左下角，右下角，因此答案一共为  $4 \times 4 = 16$  种。  
注意：不是正方形的教学楼可以有两种摆放方向；正方形教学楼只有一种方向。

## Problem F. RE: 从零开始的 NPY 生活

Input file: standard input  
Output file: standard output  
Time limit: 1 second  
Memory limit: 512 megabytes

小 P 成功找到了 NPY，成为了人生赢家。

小 P 和他的 NPY 生活在一个有  $n$  座城市的国家。城市由  $m$  条只能单向行驶的道路连接。

小 P 家住 1 号城市，小 P 的 NPY 家住  $n$  号城市，放寒假了，小 P 准备开车去他 NPY 所在的城市找她玩。

众所周知，当你去找 NPY 时，需要给她带一些礼物。

由于每个城市有独特的特产，如果小 P 决定将  $k$  号城市的特产带给他的 NPY，那么在从 1 号城市到  $n$  号城市的路上，他一定需要经过  $k$  号城市。

而小 P 并不清楚他 NPY 究竟想要什么，他决定在制定出行计划时，考虑所有可以买到的特产，并且小 P 并不想绕远。

具体来说，对于每一个  $k$ ，他需要知道，从他家所在的 1 号城市，经过  $k$  号城市，到达  $n$  号城市的所有行车方案中，路径最短的方案的路径长度。

道路交通规划部门向你保证，对于每一个城市，都有至少一种方案可以从 1 号城市到该城市再到  $n$  号城市。

这问题令小 P 十分头大，他决定求助于在 HITSZ 就读的，他的 HXD，你。希望你能够写一个程序，帮助他完成他的出行计划。

### Input

输入的第一行包含两个整数  $n(1 \leq n \leq 2000)$  和  $m(1 \leq m \leq 10^5)$ ，分别表示城市数量和单向道路的数量。对于第  $2 \sim m+1$  行，其中第  $i+1$  行有三个整数  $u_i, v_i, l_i$  表示第  $i$  条单向道路，从  $u_i$  出发，到达  $v_i$ ，长度为  $l_i$ 。保证所有的路径长度满足  $l_i \leq 2000$ 。

### Output

输出 1 行  $n$  个整数，其中第  $k$  个整数表示从 1 号城市出发，经过  $k$  号城市到达  $n$  号城市路径长度最短的路径长度。

### Example

standard input	standard output
5 9 1 2 3 2 4 4 2 3 3 1 4 2 3 5 3 3 4 2 4 1 2 4 3 1 1 5 1	1 9 6 5 1

### Note

五条可行的路径分别为  $1 \rightarrow 5$ 、 $1 \rightarrow 2 \rightarrow 3 \rightarrow 5$ 、 $1 \rightarrow 4 \rightarrow 3 \rightarrow 5$ 、 $1 \rightarrow 4 \rightarrow 1 \rightarrow 5$ 、 $1 \rightarrow 5$ 。

## Problem G. 范学长放牙刷

Input file:           standard input  
Output file:         standard output  
Time limit:          1 second  
Memory limit:       512 megabytes

众所周知，范学长早晨起来是要刷牙的。

范学长有  $N$  支牙刷，又有  $N$  个牙刷套，开始的时候，一支牙刷对应放在一个牙刷套中。可是有一天，范学长把所有牙刷套里的牙刷都拿出来，玩了一会儿，他又要把所有的牙刷都放回去。可是，他忽然一想，我可不可以使得没有任何一支牙刷放回它原来的牙刷套里面呢？

范学长努力试了很久，却一直没有成功过一次。于是他断定这个要求是无法达成的，你怎么认为的呢？

### Input

只包括一个整数  $N(1 \leq N \leq 10^5)$ ，表示牙刷和牙刷套的总数。

### Output

如果存在满足要求的方法，输出放法方案总数  $L$ 。因为方案总数可能比较大，所以请输出答案对 1206 取模后的结果。如果不存在满足要求的方法，则输出 “No Solution!”（不含引号）

### Example

standard input	standard output
3	2
4	9

### Note

对于第一组样例，易知，如果有 3 个牙刷、牙刷套，有 2 种方法使得它们重新排列并且没有任何牙刷放回原来的牙刷套。

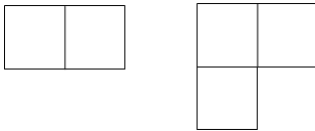
## Problem H. 芯片设计

Input file:           standard input  
Output file:         standard output  
Time limit:          1 second  
Memory limit:       512 megabytes

令人感到安稳可靠的 BangBang 学长正在参加第五届沅芯杯。

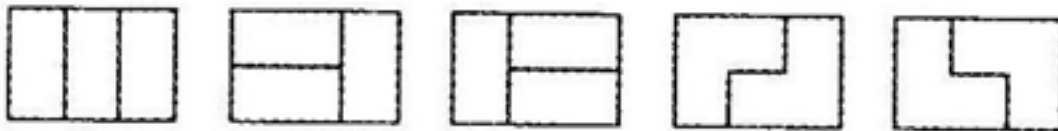
沅芯杯是一门考察学生芯片设计能力的竞赛，在本次比赛中，组委会要求选手们用给定的两种元件填满整个芯片，制造出稳定可靠的芯片的参赛选手可以获得本次比赛的一等奖。

具体来说，整个芯片可以视作一个  $2 \times N$  的矩形，而组委会给定的两种元件形状如下：

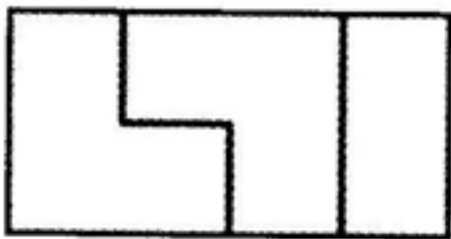


这两种元件可以旋转，两种元件可以无限制提供。设计一个稳定可靠、速度飞快的芯片对于 BangBang 学长自然不是难事，可相比于已经是囊中之物的一等奖来说，BangBang 学长更加好奇，在满足组委会要求的前提下，总共有多少种可能的芯片设计方式呢？

例如一个  $2 \times 3$  的芯片可以有 5 种芯片设计方法，如下：



注意可以使用两种元件混合起来覆盖，如  $2 \times 4$  的芯片可以这样设计：



由于结果可能很大，所以只要求出输出结果十进制下最后 4 位。例如  $2 \times 13$  的芯片的设计方法有 13465 种，所以你只需输出 3465。如果答案少于 4 位，就直接输出就可以，不用加 0，如  $N = 3$  时输出 5。

### Input

输入包含一个整数  $N(1 \leq N \leq 10^6)$ ，表示芯片的长。

### Output

输出设计方案数的最后 4 位，如果不足 4 位就输出整个答案。

### Example

standard input	standard output
3	5
13	3465



## Problem I. 玄妙排列

Input file: standard input  
Output file: standard output  
Time limit: 1 second  
Memory limit: 512 megabytes

作为一名精通集合论、图论、近世代数、组合数学等多种高端操作的优秀教师，T 老师对排列情有独钟。经过多年的潜心研究，T 老师发现，如果排列中的某一个子段满足“将这一子段排序之后，形成一段公差为 1 的等差数列”，那么这个子段将具有一种十分玄妙的特性。

(排列的子段指排列中连续的一段数字)

比如，排列 1,5,6,2,4,3 中，子段 [5,6,2,4,3] 和子段 [2,4,3] 具有玄妙的特性，而子段 [5,6,2] 则不然。

T 老师对自己的研究成果十分满意，并提出了一个更有挑战性的研究课题：

对于某个给定的排列的一个子段，该子段的所有子段中，有多少个子段具有玄妙的特性？

自然，这个课题交给了 T 老师的得意弟子 F 同学，不出所料，F 同学找到了你。

### Input

输入的第一行包含 1 个整数  $n(n \leq 10^5)$ ，表示排列的长度。

第 2 行包含  $n$  个整数  $p_i(1 \leq p_i \leq n)$ ，表示一个  $1 \sim n$  的排列。

第 3 行包含一个整数  $q(q \leq 10^5)$ ，表示询问的个数。

接下来有  $q$  行，每行两个整数  $l_i$  和  $r_i$ ，代表每次询问的排列的子段  $[l_i, r_i](1 \leq l_i \leq r_i \leq n)$ 。

### Output

输出  $q$  行，每行 1 个整数，代表询问的子段中有多少个子段具有这种特性。

### Example

standard input	standard output
5	4
1 3 2 5 4	6
3	7
2 4	
1 4	
2 5	

### Note

对于第一个询问，满足条件的子段有 [3], [2], [5], [3,2]

对于第二个询问，满足条件的子段有 [1], [3], [2], [5], [3,2], [1,3,2]

对于第三个询问，满足条件的子段有 [3], [2], [5], [4], [3,2], [5,4], [3,2,5,4]

## Problem J. 卷王之王

Input file: standard input  
Output file: standard output  
Time limit: 1 second  
Memory limit: 512 megabytes

紧张刺激的内卷环节开始了。

HITSZ 的校园里共有  $n$  位卷王，每位卷王有他自己的内卷值  $a_i$ 。

学校即将选出两名卷王凑成一队，参与第一届 HIT 双人内卷大赛。

由于学校对组队不加限制，根据 T 老师传授的组合数学相关知识，我们知道，一共有  $\frac{n \times (n-1)}{2}$  种组队方式。

当两个卷王相遇时，他们所构成的小队的内卷值并非简简单单的加和，而是由一系列复杂的因素决定。

**(此处隐藏了只有真正的卷王才知道的一系列复杂因素，成为真正的卷王后本内容自动解锁)**

由于上述种种因素，当两名内卷值分别为  $a_i$  和  $a_j$  的卷王组队时，他们构成的卷王小队的内卷值将会是  $a_i \oplus a_j$ 。HIT 的内卷冠军毫无疑问是属于 HITSZ 的，可除了这显而易见的结果，小 P 还想知道，在  $\frac{n \times (n-1)}{2}$  种组队方式中，内卷值第  $k$  大的队伍的内卷值是多少。

由于计算  $\frac{n \times (n-1)}{2}$  个小队的内卷值再排序实在令人头秃，他找到了即将代表 HITSZ 出战参与全世界大学生卷王大赛的你，他相信你一定能够给他一些帮助。

ps:  $x \oplus y$  表示数字  $x$  和  $y$  做按位异或 (Bitwise-Xor) 操作，也就是把  $x$  和  $y$  写成二进制，每一位进行异或操作，例如  $3 \oplus 5 = 6$ 。

### Input

输入的第一行包含二个正整数  $n$  ( $2 \leq n \leq 2 \times 10^5$ ) 和  $k$  ( $1 \leq k \leq \frac{n(n-1)}{2}$ )，分别表示卷王的个数以及询问的  $k$ 。第二行包含  $n$  个整数  $a_i$  ( $0 \leq a_i \leq 10^6$ )，第  $i$  个数  $a_i$  表示第  $i$  个卷王的内卷值。

### Output

输出一个整数，为生成的  $\frac{n(n-1)}{2}$  个组队中，内卷值第  $k$  大的值。

### Example

standard input	standard output
4 3 2 3 5 7	5

### Note

$a_1$  的与  $a_2 \sim a_4$  依次异或生成 1, 7, 5

$a_2$  与  $a_3 \sim a_4$  依次异或生成 6, 4

$a_3$  与  $a_4$  异或生成 2

## Problem K. ghj 的远足

Input file: standard input  
Output file: standard output  
Time limit: 2 seconds  
Memory limit: 512 megabytes

ghj 喜欢远足。

ghj 生活在一个有  $n$  个城市的国家，这  $n$  个城市由  $n - 1$  条双向通行的公路连接，并且保证，从一个城市出发，可以到达另外  $n - 1$  座城市中的任意一座城市。

寒假来了，ghj 正在设计他的远足路线。

不巧，很多城市正在举办当地的双人内卷大赛，这些城市并不准备接待游客，所以 ghj 并不能在这些城市住店休息（但是可以经过这些城市）。

由于内卷大赛的特殊赛制，赛事组委会保证，当前并没有举行内卷大赛的城市一定是标号连续的某些城市。比如 4,5,6,7 就是标号连续的 4 座城市，而 3,4,8 则不是。

在这次远足中，ghj 准备从没有举办内卷大赛的城市中挑选两座，从其中一座出发，沿公路徒步前往另一座。

ghj 真的很喜欢远足，所以他希望，在远足中，徒步经过的公路数量尽量多。

但是，内卷大赛组委会并未确定举办城市最终方案，只是给出了  $m$  种候选方案。可 ghj 等不及要开始设计线路了。

ghj 希望你对于候选方案中的每一种，都帮他设计一条路线，使得他徒步经过的公路数量尽量多。

由于 ghj 并没有 NPY，他也不关心沿路的风景与特产，所以你只需要告诉 ghj，对于每一种方案，他经过的公路数量就可以了。

### Input

输入的第一行包含两个整数  $n(2 \leq n \leq 10^5)$  和  $m(1 \leq m \leq 10^5)$ ，表示城市的数量和候选方案的数量。

接下来  $n - 1$  行，每行两个整数  $u_i, v_i$ ，表示第  $i$  条道路连接的两个城市。

接下来  $m$  行，每行两个整数  $l_i, r_i(1 \leq l_i \leq r_i \leq n)$ ，表示在第  $i$  个候选方案中，当前编号在  $[l_i, r_i]$  内的城市没有举办内卷大赛。

### Output

输出  $m$  行，第  $i$  行包含一个整数，代表对于第  $i$  个候选方案  $[l_i, r_i]$ ，ghj 最多可以徒步经过的公路数量。

### Example

standard input	standard output
6 3	2
1 2	2
2 3	3
2 4	
2 5	
4 6	
1 3	
2 4	
1 6	

## Problem L. 我不是助教

Input file:           standard input  
Output file:         standard output  
Time limit:          0.25 seconds  
Memory limit:       512 megabytes

线性代数好难鸭。

众所周知，小 L 不喜欢算矩阵乘法。

即使在他躲过了过程淘汰，成功晋升为助教之后，他也还是不喜欢矩阵乘法。

今天，小 L 需要批改学弟学妹的线性代数作业。

翻开作业本，小 L 惊奇地发现，本次作业的所有题目，都是“给定两个  $n \times n$  的方阵  $A, B$ ，求它们的乘积  $A \times B$ ”。

可学弟学妹们并不规格严格、功夫到家，因此有的答案可能有错误，小 L 需要判断学弟学妹们的答案是否正确。

由于小 L 只需要批改作业而不是做作业，他并不需要自己算矩阵乘法，他只需要验证学弟学妹的答案是否正确。

想到这里，小 L 的嘴角微微上扬，可当他看到  $n$  满足  $n \leq 1000$  时，他的笑容消失了。

三天之后，小 L 带着剩下的厚达 1.414m 的未批改的作业题，找到了你。

你看见他面色苍白，神情恍惚，你的内心产生了波动，你不忍心再让他算矩阵乘法了。

你决定写个程序帮帮他。

### Input

数据的第一行包含一个数字  $n(n \leq 1000)$ 。

接下来的  $n$  行每行  $n$  个整数，表示矩阵  $A$ 。

接下来的  $n$  行每行  $n$  个整数，表示矩阵  $B$ 。

接下来的  $n$  行每行  $n$  个整数，表示学弟学妹计算的结果矩阵  $C$ 。

保证给出的矩阵里所有数字  $|A_{ij}|, |B_{ij}| \leq 10^2, |C_{ij}| \leq 10^9$

### Output

输出一行 “Yes” 或者 “No”（不包括引号）代表学弟学妹的计算结果是否正确。

## Example

standard input	standard output
3 1 2 3 4 5 6 7 8 9 1 3 4 2 5 8 9 6 7 32 31 41 68 73 98 104 115 155	Yes
3 1 0 0 2 5 0 2 3 8 1 3 8 0 1 2 0 0 3 1 -3 8 2 -11 26 2 -9 46	No

## Note

第二个样例数据应为  $\begin{pmatrix} 1 & 3 & 8 \\ 2 & 11 & 26 \\ 2 & 9 & 46 \end{pmatrix}$

## Problem M. 过程淘汰

Input file:           standard input  
Output file:         standard output  
Time limit:          1 second  
Memory limit:       512 megabytes

一世纪规格功夫，新百年世界一流。

众所周知，哈尔滨工业大学的校训是“规格严格，功夫到家，过程淘汰”。

新的一学期里，小 P 所在的年级开设了  $N$  门课程，第  $i$  门课程的起始日期是第  $l_i$  天，终止日期是第  $r_i$  天（由于第  $r_i$  天是考试，因此课程时间包含第  $l_i$  天，但不包含第  $r_i$  天）。在这段时间里，每天都有第  $i$  门课程。

如果一个学生在一段连续时间内每天都有至少一门课，那么在这段连续的时间结束后，他会因为自己的成绩不够理想，而积累 1 点淘汰值。在整个学期结束后，由于这个学生面临找不到 npy、新年晚会抽奖没中、麦当劳队太长等不及、猫很凶不给撸等多种困难，他在这一学期的淘汰值会自我生长，变成原来的  $k$  次方。

小 P 的年级里一共有  $2^N$  个学生，每个学生的选课方案都是独一无二的。转眼快到学期末了，辅导员准备根据淘汰值来淘汰一批学生，但是，ta 首先需要知道所有  $2^N$  个人的淘汰值的和，你能帮 ta 算出来吗？

### Input

输入的第一行包含 2 个整数  $N(1 \leq N \leq 10^5)$  和  $k(1 \leq k \leq 10)$ ，表示课程数和淘汰值的生长能力。

$2 \sim N+1$  行，第  $i+1$  行有两个整数  $l_i$  和  $r_i(1 \leq l_i < r_i \leq 10^9)$  表示第  $i$  个课程的起始日期和截止日期（课程包含第  $l_i$  天，但不包含第  $r_i$  天）。

### Output

输出所有学生的淘汰值的和；由于这个数字可能很大，输出对  $10^9 + 7$  取模后的值（除以  $10^9 + 7$  的余数）。

### Example

standard input	standard output
3 3 1 6 2 3 4 5	14

### Note

对于不选任何课的同学，他的淘汰值是 0；对于同时选了第 2 和第 3 门课的同学，他的淘汰值是 8；其他六位同学淘汰值是 1；和为 14。