第三届环球杯

第 11 赛段：苏米约西

2024 年 10 月 5-6 日

该问题集应包含 15 个问题（A 至 O），共 24 页。

问题 A. 欢迎来到人大亚太中心

时间限制 4 秒 内存限制
1024 兆字节

在长度为 N 的由大写和小写英文字母组成的字符串中，求同时包含 "NPCAPC "和 "npcapc "作为子序列（不一定连续）的字符串数，模数为 998244353。

您需要解决 T 个测试案例。

制约因素

- $1 \le T \le 5000$
- $1 \le N \le 10^9$

输入

从标准输入端输入的数据格式如下

T

$_1$ 案

例$_2$

.

个案$_T$

这里，案例$_i$ 表示第 i 个测试案例。每个测试用例的格式如下：

N

输出

输出 T 行。在第 i 行，输出第 i 个测试用例的答案。

实例

标准输入	标准输出
4 12 6 5839 123456	924 0 966252995 432934749
3 123456789 987654321 999999999	333574957 124462731 163251704

备注

第一个样本案例：

在第一个测试案例中，有 924 个字符串满足条件，如 "npcapcNPCAPC "和 "NPCnpcAapPCc"。

问题 B. 子集的某些总和

时间限制 2 秒 内存限制 1024 兆字节

给你一个长度为 N 的正整数序列 $A = (A_1, A_2, \ldots, A_N)$ 。对于 $k = 0, 1, ., N$ ，求解以下问题。

> 求满足以下条件的 $\{1, 2, ., N\}$ 的子集 S 的个数。
> - 存在一个 S 的子集 T，使得 $|T| = |S| - k$ 和 $\mathrm{A}^{\sum}{}_{i \in T\, i} \geq M$ 。

制约因素

- $1 \leq N \leq 3000$
- $1 \leq M \leq 3000$
- $1 \leq A_i \leq 3000$

输入

从标准输入端输入的数据格式如下

$N\ M$
$A_1\ A_2\ .\ .A_N$

输出

输出 $N + 1$ 行。在第 i 行（$1 \leq i \leq N + 1$），输出 $k = i - 1$ 的答案。

实例

标准输入	标准输出
4 7 3 1 5 2	6 4 1 0 0
1 5 7	1 0
9 18 1 9 5 6 2 7 1 4 8	346 309 230 126 46 10 1 0 0 0

备注

第一个样本案例：

举例说明 $k = 1$ 时的情况。

- 对于 $S = \{1, 3, 4\}$，如果我们让 $T = \{3, 4\}$，那么 $|T| = |S| - 1$ 和 A $\sum_{i \in T} {}_i \geq 7$，所以它满足条件。

其他满足条件的子集有 $S = \{1, 2, 3\}$，$\{2, 3, 4\}$，$\{1, 2, 3, 4\}$，共计 3 个子集。因此，当 $k = 1$ 时，答案为 4。

问题 C. 与朋友一起解决

时间限制	2 秒 内存限制
	1024 兆字节

Namuka 和 Napuka 决定解决所有 N 个问题，即问题 1、问题 2、.问题
N .

初始时，两人的疲劳度都为 0，但解决一个问题会使解决问题的人的疲劳度增加 1。当解决第 i 个问题时，当前的疲劳度为 j，Namuka-kun 需要 $A_i + C_j$ 分钟，Napuka-kun 需要 $B_i + D_j$ 分钟。两人不能同时解决问题。

求 Namuka 和 Napuka 解决所有 N 个问题所需的最短总时间。

制约因素

- $1 \leq N \leq 2 \times 10^5$
- $1 \leq a_i , b_i , c_i , d_i \leq 10^9$

输入

从标准输入端输入的数据格式如下

N
$A_1\ A_2\ .\ .A_N$
$B_1\ B_2\ .\ .B_N\ C_0$
$C_1\ .\ .\ .C_{N-1}\ D_0$
$D_1\ .\ .\ .D_{N-1}$

输出

输出答案。

实例

标准输入	标准输出
3 1 3 5 6 4 2 1 2 3 1 2 3	10
5 2 4 3 1 2 9 2 5 3 8 1 2 8 3 2 5 4 3 2 1	28
8	621

21 85 72 22 81 20 88 28 75 22 78 92 55 56 73 44 39 14 64 27 73 42 16 84 27 7 91 85 69 95 70 27	

备注

第一个样本案例：

当 Namuka 依次解决第 1 和第 2 个问题，Napuka 解决第 3 个问题时，所花费的总时间可以计算如下：

- Namuka 解决了问题 1。Namuka 当前的疲劳度为 0，因此需要 $A_1 + C_0 = 1 + 1 = 2$ 分钟。Namuka 的疲劳度增加了 1。
- Namuka 解决了问题 2。Namuka 当前的疲劳度为 1，因此需要 $A_2 + C_1 = 3 + 2 = 5$ 分钟。Namuka 的疲劳度增加了 1。
- 纳普卡解决了问题 3。纳普卡当前的疲劳度为 0，因此需要 $B_2 + D_0 = 2 + 1 = 3$ 分钟。纳普卡的疲劳度增加 1。

因此，总时间为 $2 + 5 + 3 = 10$ 分钟，这是最少的时间。

问题 D. 两个盒子

时间限制	6 秒
内存限制	1024 兆字节

给你一个长度为 N 的非负整数序列 $A = (A_1, A_2, \ldots, A_N)$ 和 Q 个查询。其中 i-该查询的描述如下

- 将 A_{x_i} 更改为 y_i，然后根据更新后的序列 A 计算下面问题的答案。

> 有两个盒子，一白一黑，M 个球，编号从 1 到 M。一开始，所有球都在白色盒子里。
>
> 执行以下操作 N 次：
>
> - 选择一个满足 $1 \le x \le M$ 的整数 x。将小球 x 从当前方格移动到另一个方格。
>
> 在第 i 次操作之后，黑盒中所有小球上的数字必须小于或等于 A_i。计算满足这一条件的可能操作序列的数目，模数为 998244353。

按顺序处理查询。

制约因素

- $1 \le n，q \le 3 \times 10^4$
- $1 \le M \le 15$
- $1 \le x_i \le N$
- $1 \le A_i, y_i \le M$

输入

从标准输入端输入的数据格式如下

$N\ M\ Q$
$A_1\ A_2\ ..A_N$
$x_1\ y_1$
$x_2\ y_2$
$\vdots$
$x_Q\ y_Q$

输出

输出 Q 行。在第 i 行，输出第 i 个查询的答案。

实例

标准输入	标准输出
3 3 2 1 3 1 3 2 1 3	5 14
6 8 1 3 8 7 7 1 6 1 4	2100
12 10 8 1 3 2 6 3 6 7 7 5 5 4 7 12 4 7 10 4 2 9 8 9 9 8 3 4 9 10 2	2741280 3007680 1503840 1916160 1972800 728640 1821600 621440

备注

第一个样本案例：

对于第一个查询，$A = (1, 3, 2)$。在这种情况下，可能的操作序列包括：

- 选择 $x = 1$。将小球 1 从白箱移到黑箱。现在黑盒子里有小球 1。
- 选择 $x = 3$。将 3 号球从白箱移到黑箱。黑盒子里现在有球
  1 和 3。
- 选择 $x = 3$。将小球 1 从黑盒移回白盒。现在黑盒子里有 1 号球。

x 的其他可能序列是 (1，1，1)、(1，1，2)、(1，2，1) 和 (1，2，2)，总共有 4 种额外的可能性。因此，有 5 种可能的运算序列。

问题 E. 志存高远

时间限制	2 秒 内存限制
	1024 兆字节

你们将在二维平面上玩一个游戏。最初，在每个网格点 (x, y)，其中 $-100 \leq x \leq 100$ 且 $-100 \leq y \leq 0$，则放置一个棋子。

您可以执行零次或多次以下操作：

- 选择两点 (a, b) 和 (c, d)，其中 $|a-c| + |b-d| = 1$。将棋子绕 (c, d) 顺时针或逆时针旋转 90 度，移动 (a, b) 中的一个棋子，并从 (c, d) 中移走一个棋子。

你的目标是进行操作，使得在所有操作之后，至少有一个棋子位于 y *坐标*至少为 N 的点上。判断是否可能实现目标，如果可能，请构建一个操作序列。

给你 T 个测试用例。请据此解决每个测试用例。

制约因素

- $1 \leq T \leq 6$
- $1 \leq N \leq 6$

输入

从标准输入端输入的数据格式如下

T

$_1$ 案

例$_2$

.

个案$_T$

这里，案例$_i$ 表示*第* i 个测试案例。每个测试用例的格式如下：

N

输出

对于每个 T 测试用例，按给定顺序输出结果，以换行符分隔。

对于每个测试用例，如果不可能实现目标，则输出"-1"。否则，首先输出操作次数 K，然后输出描述操作的 K 行。在*第* i *次*操作中，将棋子从 (a_i, b_i) 绕 (c_i, d_i) 旋转 90 度移至 (e_i, f_i) 时，输出如下：

K

$A\ B\ C\ D\ E\ F\ A\ B$

$C\ D\ E\ F_{1\ 1\ 1\ 1\ 1}$

$_{1\ 2\ 2\ 2\ 2\ 2\ 2}$

.

.

$A\ B\ C\ D\ E\ F_{K\ K\ K\ K\ K\ K}$

示例

标准输入	标准输出
1 1	1 1 0 0 0 0 1

备注

在第一个操作中，位于（1，0）的棋子绕（0，0）顺时针旋转 90 度，然后放置在（0，1）处。

通过这一操作，可以在 y *坐标*至少为 1 的点（0，1）上放置棋子，从而实现目标。

问题 F. 火车座位

时间限制	3 秒 内存限制
	1024 兆字节

有 N 个人，从 1 到 N 依次坐在排成一排的 M 把椅子上。从左边起*第* i 把椅子叫做*第* i 把椅子。*第* i 个人坐在椅子 A *上*。$_i$

当一个人坐下来时，让 L 和 R 分别是该人左边和右边最靠近的椅子的数字（如果左边没有这样的椅子，则 $L = 0$；如果右边没有这样的椅子，则 $R = M + 1$）。该人的得分计算公式为 R - L。

N 人有 N 种可能的排序方式。求所有 N 人得分的最大可能总和。

制约因素

- $1 \le N \le 2 \times 10^5$
- $N \le M \le 10^9$
- $1 \le A_i \le M$
- 如果 $i \ne j$，那么 $A_i \ne A_j$

输入

从标准输入端输入的数据格式如下

$N\ M$
$A_1\ A_2\ . .A_N$

输出

输出答案。

实例

标准输入	标准输出
3 10 3 7 10	28
5 20 3 10 11 14 17	73
10 1000000000 136909656 243332691 <...> 182482400 (在附件中下载)	7649951260

备注

第一个样本案例：

例如，如果按照第 3 人、第 1 人和第 2 人的顺序排列，得分情况如下：

- 当第 3 人坐下时，$L = 0$，$R = 11$，因此他们的得分是 11 - 0 = 11。
- 当 1 号选手坐下时，$L = 0$，$R = 10$，因此他们的得分是 10 - 0 = 10。
- 当 2 号选手坐下时，$L = 3$，$R = 10$，因此他们的得分是 10 - 3 = 7。

因此，得分总和为 $11 + 10 + 7 = 28$，这是最大值。

问题 G. 许多常见的分段问题

时间限制：	8 秒 内存限制 1024 兆字节

PCT 带来了以下问题。

共同部分

给您 N 个片段 $[L_1, R_1], [L_2, R_2], \ldots, [L_N, R_N]$。这里，$[L, R]$ 表示从 L 到 R（含 R）的所有整数的集合。

有 $2^N - 1$ 种方法可以选择一个或多个线段，在这些方法中，求所有选择的线段的交集都不为空的方法的个数。输出结果，模数为 998244353。

PCT 意外丢失了测试用例中的一些 L_i 和 R_i 值。为了帮助他，请解决以下问题。

许多通用段测试用例

为您提供了 **Common Segment** 的测试用例。但是，缺失的 L_i, R_i 值被替换为"-1"。

已知原始测试用例满足 $1 \le L_i \le R_i \le M$ （$1 \le i \le N$）。对于所有可能的原始测试用例，求解"**公共段**"，并找出所有答案的总和，模为 998244353。

制约因素

- $1 \le N, M \le 10^5$
- $L_i = -1$ 或 $1 \le L_i \le M$
- $R_i = -1$ 或 $1 \le R_i \le M$
- 如果 $L_i, R_i \ge 1$，则 $L_i \le R_i$

输入

从标准输入端输入的数据格式如下

$N\ M$
$L_1\ R_1$
$L_2\ R_2$
$\vdots$
$L_N\ R_N$

输出

输出答案。

实例

标准输入	标准输出
3 3 1 -1 2 2 2 3	18
5 8 1 7 2 3 4 8 6 8 1 5	15
10 13 4 -1 -1 -1 7 11 -1 -1 -1 -1 -1 -1 11 -1 3 8 -1 9 -1 -1	841024210

备注

第一个样本案例：

通用段的所有可能测试用例及其相应答案如下：

- 当 $(L_i, R_i) = (1, 1), (2, 2), (2, 3)$ 时，答案为 4。
- 当 $(L_i, R_i) = (1, 2), (2, 2), (2, 3)$ 时，答案为 7。
- 当 $(L_i, R_i) = (1, 3), (2, 2), (2, 3)$ 时，答案为 7。

因此，总答案是 $4 + 7 + 7 = 18$。

问题 H. 音乐游戏

时间限制 2 秒 内存限制
1024 兆字节

有 N 个从 1 到 N 的开关。目前，所有开关都处于关闭状态。您可以按任意顺序逐个按下开关，但每个开关都是断开的。具体来说，按下开关 i 需要 T_i 秒，动作如下：

- 概率$\frac{A_i}{}$，它就会打开。
- 概率为 1 -$\frac{A_i}{}$，N 个开关全部关闭。

每次按下开关时，开关是否打开都是独立决定的。此外，在按下一个开关的同时，不能按下另一个开关。

您的目标是尽快打开所有开关。在适当按下开关后，求打开所有开关所需的预期秒数，取模为 998244353。

制约因素

- $1 \leq N \leq 2 \times 10^5$
- $1 \leq T_i \leq 10^6$
- $1 \leq a_i \leq b_i \leq 10^6$

输入

从标准输入端输入的数据格式如下

N
$T_1\ a_1\ b_1$
$T_2\ a_2\ b_2$
:
$T_N\ A_N\ B_N$

输出

可以证明，期望值总是一个有理数。此外，在这个问题的约束条件下，还可以证明当这个值表示为一个化简分数$\frac{P}{}$时，$Q \not\equiv 0$（mod 998244353）。因此，存在一个唯一的整数 $\overset{Q}{R}$，满足 $R \times Q = P$ (mod 998244353)，且 $0 \leq R < 998244353$。输出这个 R。

实例

标准输入	标准输出
2 3 3 5 2 4 7	831870305
5 2 5 9 6 4 7 1 9 14 17 8 13 10 4 11	914017655
8 6 2 8 3 1 8 5 30 71 7 9 58 6 4 7 6 9 25 2 8 67 6 6 55	923892723

备注

第一个样本案例：

下面是操作顺序的一个例子（这个顺序不一定代表最佳操作）：

- 按下开关 1 3 秒钟。开关 1 接通。
- 按 2 号开关 2 秒钟。所有开关关闭。
- 按下开关 2 2 秒钟。开关 2 接通。
- 按下开关 1 3 秒钟。开关 1 接通。

在这个序列中，所用时间为 10 秒，按此方式进行操作的概率为$\frac{3}{5}\times\frac{3}{7}\times\frac{4}{7}\times\frac{3}{5}=\frac{108}{1225}$。

此外，在这种情况下，当适当按下开关时，打开所有开关所需的预期秒数为$\frac{65}{6}$秒。

问题 I. 左等于右

时间限制	2 秒
内存限制	1024 兆字节

求满足以下条件的 $(1, \ldots, N)$ 的排列 $(P_1, \ldots, P_N)$ 的个数，并求模 998244353.

- 存在一个整数 i （$1 \le i < N$），使得 $A_{P_1} + \cdots + A_{P_i} = A_{P_{i+1}} + \cdots + A_{P_N}$。

制约因素

- $2 \le N \le 100$
- $1 \le A_i \le 100$

输入

从标准输入端输入的数据格式如下

N
$A_1\ A_2 \ldots A_N$

输出

输出答案。

实例

标准输入	标准输出
3 4 9 5	4
2 100 100	2
8 3 2 6 3 1 2 4 5	11520

备注

第一个样本案例：

1，2，3）有 3！（= 6）种排列组合，其中 4 种满足条件：

- (1, 3, 2)
- (2, 1, 3)
- (2, 3, 1)

- (3, 1, 2)

例如，对于 (1，3，2)，选择 $i = 2$，我们有 $A_1 + A_3 = A_2 = 9$，满足条件。

问题 J. 再次置换问题

时间限制 5 秒 内存限制 1024 兆字节

给你 M 个 $(1, 2, \dots, N)$ 的排列。第 i 个排列为 $P_i = (P_{i,1}, P_{i,2}, \dots, P_{i,N})$。你有一个序列 $Q = (1, 2, \dots, N)$。你可以执行下面的操作 0 次或更多次：

- 选择一个满足 $1 \le i \le M$ 的整数 i ，并将 Q 更新为 $(Q_{P_{i,1}}, Q_{P_{i,2}}, \dots, Q_{P_{i,N}})$ 。

求所有可能的序列 Q 在进行任意次数的运算后的反转数之和。输出结果，模数为 998244353。

制约因素

- $1 \le N \le 30$
- $1 \le M \le 30$
- $P_i = (P_{i,1}, P_{i,2}, \dots, P_{i,N})$ 是 $(1, 2, \dots, N)$ 的排列组合。

输入

输入内容按以下格式从标准输入端输入：

$N\ M$

P1,1 P1,2 ⋯P1,N

P2,1 P2,2 ⋯P2,N

.

$PM,1\ PM,2 \cdot PM,N$

输出

输出答案。

实例

标准输入	标准输出
3 2 1 2 3 2 3 1	4
5 2 3 4 5 1 2 1 5 4 3 2	50

30 12 1 2 9 4 5 6 <...> 26 3 28 29 30 (在附件中下载）	701414999

备注

第一个样本案例：

有三个可能的序列 Q：（1，2，3）、（2，3，1）和（3，1，2）。它们的反转数分别是 0、2 和 2，所以答案是 $0 + 2 + 2 = 4$。

问题 K. 和平与魔法

时间限制 2 秒 内存限制
1024 兆字节

NPCA 国家由直线排列的 N 个方格组成，从左到右编号为 1 到 N。假设第 i 个方格的高度为 H_i。最初，$H_1 = H_2 = \cdots = H_N = 0$。

对于每个 $1 \le i \le N-1$，如果 H_i 和 H_{i+1} 之间的绝对差小于 D_i，则第 i 个方格和第 $i+1$ 个方格之间会发生冲突。Napuka-kun 是 NPCA 国家爱好和平的国王，他的目标是消除每一对相邻方格之间的所有冲突。为了实现这一目标，Napuka-kun 可以施展以下任意次数（包括零次）的魔法：

- 选择整数 i 和 j，使得 $1 \le i \le j \le N$ 且 $H_i = H_{i+1} = \cdots = H_j$，然后在每个中加 1。$H_i, H_{i+1}, \ldots, H_j$

确定 Napuka-kun 要达到目标所需的最少魔法次数。

制约因素

- $2 \le N \le 100$
- $0 \le D_i \le 1000$

输入

从标准输入端输入的数据格式如下

N
$D_1\ D_2 \ldots D_{N-1}$

输出

输出答案。

实例

标准输入	标准输出
4 2 3 1	4
3 0 0	0
10 1 9 5 6 2 7 1 4 8	22

备注

第一个样本案例：

最初，$(H_1, H_2, H_3, H_4) = (0, 0, 0, 0)$。例如，可以按如下方式施展魔法：

- 选择 $(i, j) = (1, 3)$。则 $(H_1, H_2, H_3, H_4) = (1, 1, 1, 0)$.
- 选择 $(i, j) = (1, 2)$。则 $(H_1, H_2, H_3, H_4) = (2, 2, 1, 0)$.
- 选择 $(i, j) = (2, 2)$。则 $(H_1, H_2, H_3, H_4) = (2, 3, 1, 0)$.
- 选择 $(i, j) = (2, 2)$。则 $(H_1, H_2, H_3, H_4) = (2, 4, 1, 0)$.

纳普卡君施放 4 次魔法就能达到目标，这是最少的施放次数。请注意，您可以选择 $i = j$。

问题 L.城镇建设

时间限制 2 秒 内存限制 1024 兆字节

给你一个长度为 N - 1 的正整数非递减序列 $X = (X_1 , X_2 , ... , X_{N-1})$。定义有 N 个顶点和 M 条边的简单连通无向图 G 的代价为 $\sum_{i=1}^{N}\sum_{j=i+1}^{N} X_{d(i,j)}$。这里，$d(i, j)$ 被定义为移动所需的最小边数。从 G 中的顶点 i 到顶点 j。

构建一个有 N 个顶点和 M 条边的简单连通无向图 G，使成本最小。

制约因素

- $2 \leq N \leq 100$
- $N - 1 \leq M \leq \frac{N(N-1)}{2}$
- $1 \leq x_1 \leq x_2 \leq - - - \leq x_{N-1} \leq 10^9$

输入

从标准输入端输入的数据格式如下

$N\ M$
$X_1\ X_2\ . .X_{N-1}$

输出

当图中第 i 条边连接顶点 A_i 和顶点 B_i 时，输出 M 行如下：

$A_1\ B_1$
$A_2\ B_2$

.
$A_M\ B_M$

实例

标准输入	标准输出
3 2 4 5	1 2 1 3
4 6 12 34 56	1 2 1 3 1 4 2 3 2 4 3 4

备注

第一个样本案例：

在此输出中，成本为 $X_{d(1,2)} + X_{d(1,3)} + X_{d(2,3)} = X_1 + X_1 + X_2 = 13$。

由于不存在成本为 12 或更低的 3 个顶点和 2 条边的无向图，因此该输出是正确的。

问题 M. 敬佩的人

时间限制 2 秒 内存限制 1024 兆字节

Namuka 有一个长度为 N 的整数序列 $A = (A_1, A_2, \ldots, A_N)$，Namuka 的理想人物有一个长度为 M 的序列 $B = (B_1, B_2, \ldots, B_M)$。

为了更接近他们理想中的人，Namuka 从 A 中选择 M 个不同的元素，按任意顺序排列，形成长度为 M 的序列 $C = (C_1, C_2, \ldots, C_M)$。

找出 $\sum_{i=1}^{M} |B_i - C_i|$ 的最小可能值。

制约因素

- $1 \le m \le n \le 5000$
- $1 \le a_i, b_i \le 10^9$

输入

从标准输入端输入的数据格式如下

N M
A_1 A_2 . .A_N
B_1 B_2 . .B_M

输出

输出答案。

实例

标准输入	标准输出
5 3 2 6 5 1 1 6 3 8	4
3 2 1 1 9 1 1	0
11 7 13 21 9 5 16 32 15 29 20 40 4 24 34 43 39 18 30 11	32

备注

第一个样本案例：

例如，选择 $C = (6, 2, 5)$，可以得到最小值 $|6 - 6| + |3 - 2| + |8 - 5| = 4$。

问题 N. 产品矩阵

时间限制 3 秒 内存限制 1024 兆字节

这个问题的时限可能很紧。

给你一个 $N \times N$ 平方矩阵 $P(x)$，其中每个元素都是一级多项式。$P(x)$ 的第 (i, j) 个元素是 $a_{i,j}x + b_{i,j}$。

计算每个系数 $c_0, c_1, \ldots, c_M$ 的 (1, 1) 元素 $f(x) = \sum_{i=0}^{M} c_i x^i$ 的乘积 $\prod_{i=0}^{M-1} P(2^i x) = P(x)P(2x)\ldots P(2^{M-1}x)$, modulo $(10^9 + 7)$。

制约因素

- $1 \le N \le 6$
- $1 \le M \le 5 \times 10^5$
- $0 \le a_{i,j}, b_{i,j} < 10^9 + 7$

输入

从标准输入端输入的数据格式如下

$N\ M$

$a_{1,1}\ a_{1,2} \cdots a_{1,N}$

$a_{2,1}\ a_{2,2} \cdots a_{2,N}$

.

$a_{N,1}\ a_{N,2} \ldots\ldots a_{N,N}$

$b_{1,1}\ b_{1,2} \cdots b_{1,N}$

$b_{2,1}\ b_{2,2} \cdots b_{2,N}$

.

$b_{N,1}\ b_{N,2} \ldots b_{N,N}$

输出

输出系数 $c_0, c_1, \ldots, c_M$ modulo $(10^9 + 7)$，每个系数按此顺序各占一行。

实例

标准输入	标准输出
2 2 1 2 3 4 2 0 1 2	4 8 14

(在附件中下载)	(在附件中下载)
(在附件中下载)	(在附件中下载)

备注

第一个样本案例：

因为

$$P(x)P(2x) = \begin{pmatrix} x+2 & 2x \\ 3x+1 & 4x+2 \end{pmatrix}\begin{pmatrix} 2x+2 & 4x \\ 6x+1 & 8x+2 \end{pmatrix} = \begin{pmatrix} 14x^2+8x+4 & 20x^2+12x \\ 30x^2+24x+4 & 44x^2+28x+4 \end{pmatrix},$$

答案是$f(x) = 14x^2 + 8x + 4$。

问题 O. 新学期

时间限制 3 秒 内存限制 1024 兆字节

人大附中有 $2N$ 名学生，每个学生都有一个从 1 到 $2N$ 的唯一编号。Napuka-kun 是人大附中的一名教师，他需要把学生分成**两个班级，每个班级有** N **名学生**。

阶级划分的不满意度定义如下：

- 对于每个整数 i（$1 \le i \le M$），如果学生 A_i 和学生 B_i 在同一个班级，则在总不满意度上加上 2^i。

构建一种阶级划分方式，以尽量减少 Napuka-kun 的不满。

制约因素

- $1 \le N \le 5000$
- $0 \le M \le 10^6$
- $1 \le a_i < b_i \le 2n$
- 如果 $i \neq j$，那么 $(A_i , B_i) \neq (A_j , B_j)$
- 所有输入值均为整数

输入

输入内容按以下格式从标准输入端输入：

N M
A_1 B_1
A_2 B_2
:
A_M B_M

输出

输出格式如下

$S_1 S_2 \ldots S_{2N}$

这里，S_i 为 "0 "或 "1"，表示学生 i 属于哪个班级。

如果有多个有效的班级划分，您可以输出其中任何一个。

实例

标准输入	标准输出
2 4 1 3 2 4 1 4 1 2	0101
3 7 2 5 1 3 4 6 2 6 4 5 2 4 5 6	001101

备注

第一个样本案例：

将 1 号和 3 号学生分成一个班，将 2 号和 3 号学生分成另一个班。

4. 不满意度的计算方法如下：

- $i = 1$ 时，学生 1 和 3 在同一个班级。
- $i = 2$ 时，学生 2 和学生 4 在同一个班级。
- $i = 3$ 时，学生 1 和学生 4 分在不同的班级。
- $i = 4$ 时，学生 1 和学生 2 分在不同的班级。

因此，该除法的总不满意度为 $2^1 + 2^2 = 6$，这是最小值。您可以输出 "1010"。

如果划分为 "0111"，则不满意度为 4，但每个班级的学生人数不正好为 N，因此不满足条件。