

Problem A. 回声前缀

Input file: standard input
Output file: standard output
Time limit: 4 seconds
Memory limit: 512 megabytes

给定一个长度为 n 的小写英文字母串 S 。字符串 S 固定不变。

每个位置 i 有一个非负权值 a_i ，权值可以被修改。

对于一个非空字符串 X ，定义它的回声值为

$$H(X) = \sum_{\substack{1 \leq j \leq n-|X|+1 \\ S[j..j+|X|-1]=X}} a_j.$$

也就是说， X 在 S 中每出现一次，就贡献该出现起点的当前权值。

处理 q 次操作：

- 1 i x: 令 $a_i = x$ 。

- 2 p m: 输出

$$\sum_{t=1}^m H(S[p..p+t-1]).$$

- 3 p m k: 求最小的 t ，满足 $1 \leq t \leq m$ 且

$$\sum_{d=1}^t H(S[p..p+d-1]) \geq k.$$

若不存在这样的 t ，输出 0。

所有下标均从 1 开始。

Input

第一行包含两个整数 n, q ($1 \leq n, q \leq 2 \cdot 10^5$)。

第二行包含一个长度为 n 的小写英文字母串 S 。

第三行包含 n 个整数 $a_1, a_2, \dots, a_n$ ($0 \leq a_i \leq 10^6$)。

接下来 q 行，每行包含一次操作，格式为以下三种之一：

- 1 i x ($1 \leq i \leq n$, $0 \leq x \leq 10^6$)
- 2 p m ($1 \leq p \leq n$, $1 \leq m \leq n - p + 1$)
- 3 p m k ($1 \leq p \leq n$, $1 \leq m \leq n - p + 1$, $1 \leq k \leq 10^{18}$)

保证所有输出答案均不超过 $9 \cdot 10^{18}$ 。

Output

对于每个类型为 2 或 3 的操作，输出一行一个整数。

Example

standard input	standard output
5 9	14
ababa	3
1 0 2 1 3	34
2 1 5	3
3 1 5 10	2
1 1 5	2
2 1 5	8
3 1 5 20	
2 2 3	
1 4 4	
3 2 3 8	
2 2 3	

Note

初始时, $S = \text{ababa}$, $a = [1, 0, 2, 1, 3]$ 。

对于 $p = 1, m = 5$, 五个前缀分别为 **a**, **ab**, **aba**, **abab**, **ababa**, 它们的回声值为 6, 3, 3, 1, 1。因此第一个答案为 14。

对于询问 3 1 5 10, $t = 1, 2, 3$ 时的前缀累计和依次为 6, 9, 12, 所以答案为 3。

将 $a_1 = 5$ 后, 上述五个回声值变为 10, 7, 7, 5, 5, 所以 2 1 5 的答案为 34。

Problem B. 放大的徽章

Input file: standard input
Output file: standard output
Time limit: 3 seconds
Memory limit: 512 megabytes

平面上给定两个有限格点集合 A 和 B 。设 $P = \text{conv}(A)$, $Q = \text{conv}(B)$ 分别为它们的凸包。输入中可能包含重复点。

对于一个正整数 k , 将 P 关于原点放大 k 倍, 得到多边形 kP 。

如果整数向量 (u, v) 使得平移后的多边形 $kP + (u, v)$ 与 Q 至少有一个公共点, 则称 (u, v) 是合法的。边界相交也算相交。

对于每个询问 k , 求合法整数向量 (u, v) 的数量, 并对 998244353 取模。

Input

第一行包含三个整数 n, m, q ($3 \leq n, m \leq 2 \cdot 10^5$, $1 \leq q \leq 2 \cdot 10^5$), 分别表示 A 中点的个数、 B 中点的个数和询问次数。

接下来 n 行, 每行包含两个整数 x_i, y_i ($-10^9 \leq x_i, y_i \leq 10^9$), 表示 A 中的一个点。

接下来 m 行, 每行包含两个整数 x_i, y_i ($-10^9 \leq x_i, y_i \leq 10^9$), 表示 B 中的一个点。

接下来 q 行, 每行包含一个整数 k ($1 \leq k \leq 10^{18}$), 表示一次询问。

保证 P 和 Q 的面积均为正。

Output

对于每个询问, 输出一行一个整数, 表示合法整数向量的数量对 998244353 取模的结果。

Example

standard input	standard output
6 5 4	20
0 0	36
2 0	108
2 1	308
0 1	
1 0	
1 1	
0 0	
3 0	
1 2	
0 1	
1 1	
1	
2	
5	
10	

Note

在样例中, P 是顶点为 $(0,0), (2,0), (2,1), (0,1)$ 的矩形, Q 是顶点为 $(0,0), (3,0), (1,2), (0,1)$ 的四边形。

可以算出 $2S_P = 4$, $2S_Q = 7$, $B_P = 6$, $B_Q = 7$, 且 $2S_{Q+(-P)} = 25$ 。因此询问 k 的答案为 $2k^2 + 10k + 8$ 。

代入 $k = 1, 2, 5, 10$, 答案分别为 20, 36, 108, 308。

Problem C. 数字游戏

Input file: standard input
Output file: standard output
Time limit: 1 seconds
Memory limit: 256 megabytes

给定一个正整数进制 B 。

你需要构造三个长度为 B 的排列 P, Q, R （即每个序列恰好包含 0 到 $B-1$ 每个数字一次）。

$$P = (p_1, p_2, \dots, p_B), \quad Q = (q_1, q_2, \dots, q_B), \quad R = (r_1, r_2, \dots, r_B),$$

定义一个序列 $X = (x_1, x_2, \dots, x_B)$ 在 B 进制下表示的整数为

$$[X]_B = \sum_{i=1}^B x_i \cdot B^{B-i}.$$

注意序列允许以数字 0 开头，因此表示的整数可以小于 B^{B-1} 。

请你构造三个排列 P, Q, R ，使得它们满足等式（注意加法是在 B 进制下的加法）

$$[P]_B + [Q]_B = [R]_B,$$

并且对于每个位置 i ($1 \leq i \leq B$)， p_i, q_i, r_i 两两不同。

如果不存在这样的构造，请输出 -1 。

Input

输入仅一行，包含一个整数 B ($2 \leq B \leq 10^6$)。

Output

如果存在合法构造，则输出三行，每行 B 个整数，分别表示序列 P, Q, R 中的数字，数字之间用空格分隔。

如果不存在合法构造，则输出一行 -1 。

若存在多种构造，输出任意一种即可。

Example

standard input	standard output
5	-1
6	4 2 5 0 3 1 0 4 1 2 5 3 5 1 0 3 2 4

Note

$2 \leq B \leq 10^6$ 。

Problem D. 半平面覆盖

Input file: standard input
Output file: standard output
Time limit: 1 seconds
Memory limit: 1024 megabytes

给定若干组测试数据。每组测试数据中，平面上有 n 个闭半平面。

第 i 个半平面由三个整数 a_i, b_i, c_i 描述，包含所有满足

$$a_i x + b_i y + c_i \geq 0$$

的点 (x, y) 。

对每组测试数据，请选择其中一些半平面，使它们的并集为整个平面。也就是说，任意点 (x, y) 都必须属于至少一个被选择的半平面。

判断是否存在这样的选择方案。若存在，输出选择半平面数量最少的一种方案。

Input

第一行包含一个整数 t ($1 \leq t \leq 5000$)，表示测试数据组数。

对于每组测试数据，第一行包含一个整数 n ($1 \leq n \leq 5000$)，表示半平面个数。

接下来 n 行，每行包含三个整数 a_i, b_i, c_i ($-10^6 \leq a_i, b_i, c_i \leq 10^6$)，描述一个半平面。

对于每个 i ，保证 a_i 和 b_i 不同时为 0。

保证所有测试数据的 n 之和不超过 5000。

Output

对每组测试数据，独立输出答案。

如果不存在合法选择方案，输出一行 NO。

否则，第一行输出 YES。

第二行输出一个整数 k ，表示被选择半平面的最少数目。

第三行输出 k 个两两不同的整数，表示被选择半平面的下标。下标在每组测试数据内从 1 开始。

如果存在多种最优答案，输出任意一种即可。

Example

standard input	standard output
3 2 1 0 0 -1 0 0 3 1 0 0 0 1 0 -1 -1 0 3 1 0 -1 0 1 -1 -1 -1 -1	YES 2 1 2 YES 3 1 2 3 NO
1 5 1 0 -100 0 1 -100 1 1 -1000 2 2 -2 -3 -3 6	YES 2 4 5
1 4 -1000000 -1000000 -1000000 1000000 -1000000 1000000 1000000 1000000 -1000000 -1000000 999999 1000000	YES 3 2 3 4

Note

在第一组测试数据中，两个半平面分别为 $x \geq 0$ 和 $x \leq 0$ ，它们的并集为整个平面。

在第二组测试数据中，三个半平面分别为 $x \geq 0$ 、 $y \geq 0$ 和 $x + y \leq 0$ 。任意两个都无法覆盖整个平面，但三个一起可以。

在第三组测试数据中，点 (0,0) 不属于任何给定半平面，因此无法覆盖整个平面。

半平面是闭半平面，边界上的点也属于半平面。

样例 2 中，最后两个半平面可以覆盖整个平面，前三个半平面是干扰项，不能构成更小的合法覆盖。

样例 3 用于检验系数接近上界时的精确整数运算。

Problem E. 古明地恋与非积辈

Input file: standard input
Output file: standard output
Time limit: 1 seconds
Memory limit: 256 megabytes

古明地恋正在学习数论！

在数论领域，我们称一个函数 $f(n)$ 是**完全积性函数**，当且仅当：

$$\forall a \cdot b = n, \text{ s.t. } f(n) = f(a) \cdot f(b)$$

这种积性这对 n 的每个非平凡的分解也都成立。也就是说，

$$\forall n = w_1 \cdot w_2 \cdots w_k, 1 < w_i < n, \text{ s.t. } f(n) = f(w_1) \cdot f(w_2) \cdots f(w_k)$$

由于古明地恋不喜欢积性，于是她提出了一个**完全非积性函数** $g(n)$ ，希望满足：

$$\forall n = w_1 \cdot w_2 \cdots w_k, 1 < w_i < n, \text{ s.t. } g(n) \neq g(w_1) \cdot g(w_2) \cdots g(w_k)$$

为了让问题更简单，古明地恋做了如下定义：

$$\begin{cases} g(n) = 1 & n = 1 \\ g(n) = \text{mex} \left(\left\{ \prod g(w_i) \mid n = \prod w_i, 1 < w_i < n \right\} \cup \{0\} \right), & n = 2, 3, 4, \dots \end{cases}$$

其中 $\text{mex}(S)$ 表示集合 S 中未出现过的最小的自然数。

由于古明地恋是无意识少女，她希望你帮她计算 g 函数的值。具体而言，古明地恋会给你整数 N 以及参数 c ($1 \leq N \leq 10^7$, $1 \leq c < 998,244,353$)，你需要告诉她：

$$\text{ans} = \left(\sum_{i=1}^N g(i) \times c^i \right) \bmod 998,244,353$$

Input

第一行包含两个整数 N, c ($1 \leq N \leq 10^7$, $1 \leq c < 998244353$)。

Output

输出一个整数，表示答案。

Example

standard input	standard output
8 10	312121110
114 514	265738076

Note

Problem F. 四

Input file: **standard input**
Output file: **standard output**
Time limit: 4 seconds
Memory limit: 512 megabytes

给定四个正整数 A, B, C, D 以及四个非负整数 p_1, p_2, p_3, p_4 ，你需要计算

$$\sum_{i=1}^A \sum_{j=1}^B \sum_{k=1}^C \sum_{l=1}^D i^{p_1} \cdot j^{p_2} \cdot k^{p_3} \cdot l^{p_4} \cdot \text{lcm}(\text{gcd}(i, j), \text{gcd}(j, k), \text{gcd}(k, l))$$

对 998244353 取模后的值。

其中 $\text{gcd}(x, y)$ 表示 x 和 y 的最大公约数， $\text{lcm}(x, y, z)$ 表示 x, y, z 的最小公倍数。

Input

输入仅一行，包含八个整数，依次为 $A, B, C, D, p_1, p_2, p_3, p_4$ ($1 \leq A, B, C, D \leq 2 \times 10^6$, $0 \leq p_1, p_2, p_3, p_4 \leq 10^9$)。

Output

输出一个整数，表示所求和对 998244353 取模的结果。

Example

standard input	standard output
1 1 1 2 0 1 2 3	9
3 4 5 6 2 3 5 7	427375642

Note

$1 \leq A, B, C, D \leq 2 \times 10^6$, $0 \leq p_1, p_2, p_3, p_4 \leq 10^9$ 。

Problem G. 二分图

Input file: standard input
Output file: standard output
Time limit: 6 seconds
Memory limit: 1024 megabytes

给定一个二分图 G ，其左部点集为 L ，大小为 n ；右部点集为 R ，大小为 m 。边集为 E ，规定左部点的编号为 $1 \sim n$ ，右部点的编号为 $n+1 \sim n+m$ 。

令 $\mathcal{S}$ 表示 G 的所有**最小点覆盖**所构成的集合（每个最小点覆盖是顶点集的一个子集）。

现给出 q 次独立的询问。每次询问包含若干条"要求"，每条要求形如：

- $u\ 0$ ，要求点 u 必须被选中（即属于所考虑的最小点覆盖）；
- $u\ 1$ ，要求点 u 一定不能被选中（即不属于所考虑的最小点覆盖）。

对于每次询问，你需要判断：是否存在一个最小点覆盖 $P \in \mathcal{S}$ ，使得 P 满足该次询问中给出的所有要求。

如果存在，你还需要进一步计算：

1. 在所有满足本次询问所有要求的那些最小点覆盖 P 中，**必定会被选中的**顶点数量（即满足要求的最小点覆盖的交集大小）；
2. 在所有满足本次询问所有要求的那些最小点覆盖 P 中，**必定不会被选中的**顶点数量（即满足要求的最小点覆盖的并集的补集大小，或者说在所有这些 P 中都不出现的顶点个数）。

Input

第一行包含四个整数 n, m, e, q （ $1 \leq n, m \leq 10^5$ ， $0 \leq e \leq \min(3 \times 10^5, nm)$ ， $1 \leq q \leq 2 \times 10^5$ ），分别表示左部点数、右部点数、边数和询问次数。

接下来 e 行，每行两个整数 u, v ，表示左部点 u 与右部点 v 之间有一条边。

之后输入 q 次询问。每次询问的第一行包含一个整数 t ，表示要求数量；接下来 t 行，每行两个整数 x, y ，其中 x 为顶点编号， $y \in \{0, 1\}$ 。 $y = 0$ 表示该点必须被选中， $y = 1$ 表示该点不能被选中。

所有询问的要求总数不超过 3×10^5 。

Output

对于每次询问，若不存在满足要求的最小点覆盖，一行输出一个 -1 ，否则一行包含两个整数，分别表示**必定会被选中的**顶点数量，**必定不会被选中的**顶点数量。

Example

standard input	standard output
3 5 15 5	3 5
1 4	3 5
1 5	-1
1 6	-1
1 7	-1
1 8	
2 4	
2 5	
2 6	
2 7	
2 8	
3 4	
3 5	
3 6	
3 7	
3 8	
2	
1 0	
4 1	
2	
1 0	
1 0	
2	
3 1	
8 0	
2	
6 1	
1 1	
2	
8 0	
8 0	

Note

$1 \leq n, m \leq 10^5$, $0 \leq e \leq \min(3 \times 10^5, nm)$, $1 \leq q \leq 2 \times 10^5$, 所有询问中要求的总条数不超过 3×10^5 。

Problem H. 数计

Input file: standard input
Output file: standard output
Time limit: 1 seconds
Memory limit: 256 megabytes

给定一个长度为 N 的整数序列 $A_1, A_2, \dots, A_N$ ，其中每个元素满足 $0 \leq A_i < M$ 。

对于每个切分点 i ($1 \leq i < N$)，定义：

— 前缀集合 $S_i = \{A_1, A_2, \dots, A_i\}$;

— 后缀集合 $T_i = \{A_{i+1}, A_{i+2}, \dots, A_N\}$ 。

令

$$P_i = \text{mex}(S_i), \quad Q_i = \text{mex}(T_i), \quad C_i = \min(P_i, Q_i),$$

其中 $\text{mex}(X)$ 表示集合 X 的 mex，即最小的非负整数 x ，满足 $x \notin X$ 。

现在给定一个长度为 $N-1$ 的数组 $B_1, B_2, \dots, B_{N-1}$ ，你需要统计有多少个长度为 N 的序列 A （满足 $0 \leq A_i < M$ ）使得对于所有 $1 \leq i < N$ ，都有 $C_i = B_i$ 。

答案对 998244353 取模。

Input

第一行包含两个整数 N, M ($2 \leq N \leq 5000$, $1 \leq M \leq 10^{18}$)。

第二行包含 $N-1$ 个整数 $B_1, B_2, \dots, B_{N-1}$ ($0 \leq B_i \leq M$)。

Output

输出一个整数，表示满足条件的序列 A 的数量对 998244353 取模后的结果。

Example

standard input	standard output
4 2 1 2 1	1
5 2 1 1 2 1	1
10 10 0 0 0 0 6 0 0 0 0	0
10 100 0 1 2 2 2 2 2 2 0	279689447

Note

$2 \leq N \leq 5000$, $1 \leq M \leq 10^{18}$, $0 \leq B_i \leq M$ 。

Problem I. 题列序 2

Input file: standard input
Output file: standard output
Time limit: 1 seconds
Memory limit: 256 megabytes

给定一个整数 n ，令 $N = 2^n - 1$ 。你有一个长度为 N 的 01 串 s ，下标从 1 到 N 。

现在依次进行 N 个操作，操作编号为 $x = 1, 2, \dots, N$ 。对于第 x 个操作，你可以选择以下两种行动之一：

- **跳过**：不进行任何操作；
- **执行翻转**：选择一对整数 (y, z) ，满足 $1 \leq y, z \leq N$ ， $y \neq z$ ，且 $x = y \oplus z$ ，其中 $\oplus$ 表示按位异或运算。然后同时翻转位置 x, y, z 上的二进制位（即 0 变 1，1 变 0）。

你需要给出一种操作方案，使得最终整个二进制串中至多只有一个位置上的值为 1。输出时，若执行第 x 个操作，只需给出 y ，而 z 由 $z = x \oplus y$ 唯一确定。可以证明一定有解。

Input

第一行包含一个整数 T ($1 \leq T \leq 1000$)，表示测试数据组数。

每组测试数据包含两行：第一行包含一个整数 n ($1 \leq n \leq 20$)；第二行包含一个长度为 $N = 2^n - 1$ 的 01 串 s ，表示初始二进制串。

保证所有测试数据的 $\sum N \leq 2 \times 10^6$ 。

Output

对于每组测试数据，输出一行 N 个整数 $b_1, b_2, \dots, b_N$ ，其中 b_x 描述第 x 个操作：

若不进行操作，令 $b_x = 0$ ；

否则，令 $y = b_x, z = x \oplus y$ ，并执行对位置 x, y, z 的翻转。此时必须满足 $1 \leq y, z \leq N$ 。

Example

standard input	standard output
8	0 0 0
2	0 0 0
000	0 0 0
2	0 1 0
100	0 0 0
2	0 0 1
010	0 3 0
2	0 3 0
110	
2	
001	
2	
101	
2	
011	
2	
111	

Note

$1 \leq T \leq 1000$, $1 \leq n \leq 20$, $\sum N \leq 2 \times 10^6$ 。

Problem J. 序列变换 (median version)

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

有一个长度为 n 的环状整数序列 $a_0, a_1, \dots, a_{n-1}$ 。定义一次 **变换** 如下:

1. 根据当前序列 a 构造一个新序列 $b_0, b_1, \dots, b_{n-1}$, 其中

$$b_i = \text{median}\{a_i, a_{(i+1) \bmod n}, a_{(i+n-1) \bmod n}\}$$

2. 将序列 a 替换为 b 。

这里 $\text{median}(x, y, z)$ 表示三个数中的中位数, 即去掉最大值和最小值后剩下的那个数。

现在给定初始序列 a 和变换次数 k , 请你求出经过 k 次变换后得到的序列。

Input

第一行包含两个整数 n 和 k ($1 \leq n \leq 10^6$, $1 \leq k \leq 10^9$)。

第二行包含 n 个整数 $a_0, a_1, \dots, a_{n-1}$ ($0 \leq a_i \leq 10^9$), 表示初始序列。

Output

输出一行 n 个整数, 表示最终序列, 数与数之间用一个空格隔开。

Example

standard input	standard output
3 1 0 1 2	1 1 1
4 1 0 1 2 3	1 1 2 2
4 2 0 2 0 2	0 2 0 2
10 4 0 1 2 3 4 2 1 0 5 6	1 1 2 3 3 2 1 1 5 5
12 6 12 9 7 8 1 2 4 5 6 7 14 20	12 9 8 7 2 2 4 5 6 7 14 14

Note

Problem K. 序列变换 (mex version)

Input file: standard input
Output file: standard output
Time limit: 2 seconds
Memory limit: 256 megabytes

有一个长度为 n 的环状整数序列 $a_0, a_1, \dots, a_{n-1}$ 。定义一次 **变换** 如下:

1. 根据当前序列 a 构造一个新序列 $b_0, b_1, \dots, b_{n-1}$, 其中 $b_i = \text{mex}\{a_i, a_{(i+1) \bmod n}, a_{(i+n-1) \bmod n}\}$ 。这里 mex 表示一个非负整数集合中最小的没有出现的非负整数 (例如 $\text{mex}\{0, 1, 3\} = 2$)。
2. 将序列 a 替换为 b 。

现在给定初始序列 a 和变换次数 k , 请你求出经过 k 次变换后得到的序列。

Input

第一行包含两个整数 n 和 k ($1 \leq n, k \leq 10^6$)。

第二行包含 n 个整数 $a_0, a_1, \dots, a_{n-1}$ ($0 \leq a_i \leq 10^9$), 表示初始序列。

Output

输出一行 n 个整数, 表示最终序列, 数与数之间用一个空格隔开。

Example

standard input	standard output
4 1 0 1 2 3	2 3 0 1
4 2 0 2 0 2	0 0 0 0

Note

Problem L. 又一个矩阵问题

Input file: standard input
Output file: standard output
Time limit: 1 seconds
Memory limit: 1024 megabytes

给定一个大小为 $n \times m$ 的整数矩阵 a 。

你可以进行任意多次如下操作：选择一条从 $(1,1)$ 出发到达 (n,m) 的路径，每次移动只能向右或向下，然后将该路径经过的所有格子上的数值增加 1。

你的目标是使矩阵中所有元素的值均相等。

请求出最少需要的操作次数；如果无法达成目标，请输出 -1 。

Input

第一行包含一个整数 T ($1 \leq T \leq 5000$)，表示测试数据组数。

每组测试数据的第一行包含两个整数 n, m ，表示矩阵的行数和列数。接下来 n 行，每行包含 m 个整数，表示矩阵 a ，其中第 i 行第 j 列的元素为 $a_{i,j}$ ($-10^9 \leq a_{i,j} \leq 10^9$)。

保证所有测试数据满足 $1 \leq \sum n, \sum m \leq 5000$ 。

Output

对于每组测试数据，输出一行一个整数，表示该组数据的最少操作次数。若无法使矩阵中所有数相等，则输出 -1 。

Example

standard input	standard output
2	1
2 3	2
1 1 1	
2 2 1	
3 2	
1 2	
2 2	
2 1	

Note

$1 \leq T \leq 5000$, $1 \leq \sum n, \sum m \leq 5000$, $-10^9 \leq a_{i,j} \leq 10^9$ 。

Problem M. 火漆

Input file: standard input
Output file: standard output
Time limit: 5 seconds
Memory limit: 512 megabytes

*Je t'aime encore
Mais le monde pleure sans fin*

— MONEY, 《Je t'aime encore》

一棵**满三叉平面树**是指一棵有根有序树，并且每个内部结点恰好有 3 个儿子。

定义每个结点的权值如下：

1. 叶子结点的权值为 0；
2. 若一个内部结点的三个儿子的权值分别为 a, b, c ，令 $m = \max\{a, b, c\}$ ，若 a, b, c 中等于 m 的值至少出现两次，则该结点权值为 $m + 1$ ，否则该结点权值为 m 。

现在给定整数 H, N, K 。

考虑所有满足以下条件的满三叉平面树：

1. 整棵根结点的权值恰好为 H ；
2. 内部结点数恰好为 $2^H - 1 + N$ 。

对于这样一棵树，将所有权值等于 H 的内部结点都染成蓝色。

对每个 $1 \leq k \leq K$ ，你需要求出蓝色结点数恰好为 k 的合法树数量对 2 取模后的值。

Input

第一行包含一个整数 T ($1 \leq T \leq 5$)，表示测试数据组数。

接下来 T 行，每行包含三个整数 H, N, K ($1 \leq H \leq 10^{18}$, $1 \leq K \leq N + 1$)，含义如上所述。

保证所有测试数据满足 $0 \leq \sum N \leq 5 \times 10^5$ 。

Output

对于每组测试数据，输出一行长度为 K 的 01 串。

第 k 个字符 ($1 \leq k \leq K$) 表示：满足条件的树中，蓝色结点数恰好为 k 的树的个数对 2 取模后的值（即若为奇数则输出 1，偶数则输出 0）。

Example

standard input	standard output
1 2 10 11	01110001011
1 3 3 4	0011
1 20 500000 17	000000000000000000

Note

$1 \leq T \leq 5$, $1 \leq H \leq 10^{18}$, $0 \leq \sum N \leq 5 \times 10^5$, $1 \leq K \leq N + 1$ 。

Problem N. 欢迎来到哈尔滨工业大学

Input file: **standard input**
Output file: **standard output**
Time limit: 1 seconds
Memory limit: 256 megabytes

许多大学通常使用大写英文缩写。例如，清华大学的英文缩写是 THU，北京大学的英文缩写是 PKU。哈尔滨工业大学的英文名称是 Harbin Institute of Technology。请输出它的大写英文缩写。

Input

本题没有输入。

Output

输出 Harbin Institute of Technology 的大写英文缩写。

Example

standard input	standard output
—	THU

Note

样例输出仅用于展示输出格式，不一定是本题的正确答案。