

Problem A. 最小乘积

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

小 A 有一个包含 N 个点和 M 条边的有向图。对于第 i 条边 (u_i, v_i) ，有两个属性 (a_i, b_i) 。
定义路径 P 为从点 1 到点 N 的一条有向路径，其路径权值为：

$$\left(\sum_{i \in P} a_i\right) \times \left(\sum_{i \in P} b_i\right)$$

小 A 很想知道从节点 1 到节点 N 的所有可能路径中，最小的权值是多少。请你帮助他计算出这个最小的乘积权值。

Input

第一行一个正整数 T ($1 \leq T \leq 10^3$)，表示数据组数。
对于每组数据，第一行输入两个整数 N, M ($1 \leq N \leq 300, 1 \leq M \leq 10^3$)，分别表示图的点数和边数。
接下来的 M 行，每行包含四个整数 u_i, v_i, a_i, b_i ($1 \leq u_i, v_i \leq N, u_i \neq v_i, 1 \leq a_i, b_i \leq 200$)，分别表示有向图中一条边的起点、终点及其权值属性。
保证至少存在一条 1 号点到 N 号点的路径。
保证单个测试点内每组数据中 N 的和不超过 10^3 ， M 的和不超过 2×10^3 。

Output

对于每组数据，输出路径权值最小的路径所对应的 $\sum_{i \in P} a_i$ 和 $\sum_{i \in P} b_i$ 的值。
如果存在多条路径具有相同的最小权值，请输出其中 $\sum_{i \in P} a_i$ 最小的一条对应的值。

Example

standard input	standard output
1 5 9 3 4 3 5 4 5 5 1 1 4 2 2 3 4 5 2 1 4 2 4 2 1 3 2 4 2 5 4 4 1 2 2 4 1 3 1	7 3

Problem B. 三进制

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

这是一道交互题。

对于 n 位三进制数，小 A 设计了一个非常简单的加密算法：对每一位 $i = 0, 1, \dots, n-1$ 构造双射 $f_i: \{0, 1, 2\} \rightarrow \{0, 1, 2\}$ 。对于一个 n 位三进制数 A ，假设其加密前从高位到低位依次是：

$$a_{n-1}, a_{n-2}, \dots, a_0$$

那么其加密后从高位到低位就是：

$$f_{n-1}(a_{n-1}), f_{n-2}(a_{n-2}), \dots, f_0(a_0)$$

为了方便表示每一位的加密，我们可以将双射 $f_i = \{0 \rightarrow x, 1 \rightarrow y, 2 \rightarrow z\}$ （当然有 $x, y, z \in \{0, 1, 2\}$ 且 x, y, z 互不相同）简单地表示为 $f_i = xyz$ 。例如 $f_0 = \{0 \rightarrow 1, 1 \rightarrow 2, 2 \rightarrow 0\}$ 就可以表示为 $f_0 = 120$ 。

比如小 A 对 3 位三进制数构造双射 $f_0 = 021, f_1 = 120, f_2 = 210$ ，那么 120 加密后就是 100，201 加密后就是 012。

小 A 当然不希望别人知道它构造的双射，但是加密算法终究是要使用的，所以小 A 会告诉你 n 的大小，并允许你询问 2 次加密状态下加法的结果，而你的任务是依靠这 2 次询问破解小 A 构造的 n 个双射 f_0, f_1, \dots, f_{n-1} 。

Input

第一行一个正整数 T ($1 \leq T \leq 10^4$)，表示数据组数。

每组数据一行一个正整数 n ($1 \leq n \leq 10^5$)，表示三进制位数。

保证单个测试点内每组数据中 n 的和不超过 10^6 。

Interaction Protocol

你可以进行不超过 2 次询问，之后回答小 A 的加密映射。

每次询问，你应该以格式 `? a b` 给出加法的两个参数 a, b ，其中 a, b 都是 n 位三进制数并要求你以字符串格式给出。在刷新输出流之后，你需要以字符串格式读入一行一个 $n+1$ 位三进制数 c ，其中 c 是 a, b 分别解密后相加之后再加密的结果，其中最高位作为溢出标志不加密。形式化的：

$$c = f(f^{-1}(a) + f^{-1}(b))$$

其中 f 是加密双射， f^{-1} 是 f 的逆映射。

注意，所有三进制字符串的输入都是从高位到低位！

注意， c 的最高位即第 n 位作为溢出标志不加密，或者你可以认为 $f_n = 012$ 。

如果要回答小 A 的加密映射，你应该以格式 `! f_0 f_1 \dots f_{n-1}` 给出 n 个加密映射，其中 f_0, f_1, \dots, f_{n-1} 都应该是字符串 012, 021, 102, 120, 201, 210 中的一个。在刷新输出流之后，交互器会立即判定你的答案是否正确，然后进行下一组数据交互或结束程序，而不会有多余输出。

注意，小 A 的加密映射在询问前已经完全确定，不会随着询问更改。换句话说，交互器不是自适应的。

为了刷新输出流，你可以：

- 在 C/C++ 中使用 `fflush(stdout)`（如果使用 `printf`）或者 `cout.flush()`（如果使用 `cout`）。
- Java 中使用 `System.out.flush()`。

- 在 Python 中使用 `sys.stdout.flush()`。

Example

standard input	standard output
2	? 011 102
3	? 010 202
0210	! 021 120 102
1102	? 0 1
1	! 012
01	

Note

对于第一组数据的第一个询问，011 和 102 解密后分别是 102 和 021，相加后得到 0200，再加密后是 0210。

对于第一组数据的第二个询问，010 和 202 解密后分别是 100 和 221，相加后得到 1021，再加密后是 1102。

对于第二组数据的第一个询问，0 和 1 解密后分别是 0 和 1，相加后得到 01，再加密后是 01。

Problem C. 最优时间

Input file: `standard input`
Output file: `standard output`
Time limit: 3 seconds
Memory limit: 256 megabytes

定义集合 $S(x)$ 为:

$$S(x) = \{d \mid d \mid x\} \cup \left\{ kx \mid 2 \leq k \leq \left\lfloor \frac{N}{x} \right\rfloor \right\}$$

简单地说, $S(x)$ 是 x 的所有约数和所有不超过 N 的倍数的并集。
假设当前你的状态是 x , 每一秒, 你有两种选择:

- 等概率变成 $S(x)$ 中的一个数字。
- 什么都不做。

在每秒结束后, $x \leftarrow x - 1$ 。
求在最优决策下走到 0 状态的期望时间。

小 A 想知道针对不同的 x , 最优决策下 x 走到 0 的期望时间, 所以他有 Q 次询问, 请你帮助他回答这些问题。

Input

第一行两个正整数 N, Q ($1 \leq N, Q \leq 10^5$), 分别表示值域大小和询问次数。
接下来 Q 行, 每行一个正整数 x ($1 \leq x \leq N$), 表示一次询问。

Output

对于每个询问, 你需要输出一个实数。
如果和正确答案的绝对误差或相对误差不超过 10^{-6} , 则认为是正确答案。

Examples

standard input	standard output
3 2 3 2	1.7500000000 1.5000000000
1000 2 114 514	4.7506538205 3.7750763456

Problem D. 三分图

Input file: `standard input`
Output file: `standard output`
Time limit: 2 seconds
Memory limit: 1024 megabytes

小 M 得到了一张三分图。
一张无向图是三分图，当且仅当存在一种方案，可以给图中每个点染成 1, 2, 3 三种颜色之一，使得图中每条边连接的两个点的颜色均不相同。
对于一个长度为 n 的全排列 p ，小 A 通过以下方式生成一张 n 个点的图：
对于 $1 \leq i < j \leq n$ ，如果 $p_i > p_j$ ，则在图中连接一条无向边 (i, j) ，否则图中不存在无向边 (i, j) 。
现在给出一个长度为 n 的全排列 q ，请问有多少个长度为 n 的全排列 p ，满足 p 的字典序大于 q ，且由 p 生成的图为三分图，答案对 998244353 取模。

Input

第一行一个正整数 T ($1 \leq T \leq 300$)，表示数据组数。
对于每组数据，第一行一个正整数 n ($1 \leq n \leq 300$)，表示排列长度。
第二行 n 个整数 q_1, q_2, \dots, q_n ($1 \leq q_i \leq n$)，意义如题。保证 $i \neq j$ 时 $q_i \neq q_j$ ，即 q 是排列。

Output

对于每组数据，输出一行一个整数表示答案，答案对 998244353 取模。

Example

standard input	standard output
2	102
5	96
1 2 3 4 5	
5	
1 3 2 4 5	

Problem E. 竞赛图

Input file: **standard input**
Output file: **standard output**
Time limit: 3 seconds
Memory limit: 256 megabytes

小 A 得到了一张竞赛图。
给定 n, k ，求有多少个 n 个点的有标号竞赛图，图中至少存在一个 k 元环，答案对 998244353 取模。
请和小 A 一起来回忆以下概念：
竞赛图： n 个点， $\frac{n(n-1)}{2}$ 条边的有向图，其中任意两个编号不同的点之间都恰好存在一条有向边。
 k 元环： 长度为 k 的两两不同的点序列 v_1, v_2, \dots, v_k ，图中存在有向边：

$$(v_1, v_2), (v_2, v_3), \dots, (v_{k-1}, v_k), (v_k, v_1)$$

Input

第一行给定两个正整数 n, k ($2 \leq n, k \leq 10^5$)。

Output

输出一个整数表示答案，答案对 998244353 取模。

Examples

standard input	standard output
3 2	0
3 3	2

Problem F. 逆序对

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

有一个长度为 n 的整数序列 a_1, a_2, \dots, a_n ，序列中的整数只有 0 和 1，但序列中有些位置上的元素还没有被确定。

现在你想往序列中未被确定的位置填入 0 或者 1，使得其逆序对数最大。

一个序列 a_1, a_2, \dots, a_n 的逆序对数定义为，满足 $1 \leq i < j \leq n$ 并且 $a_i > a_j$ 的整数数对 i, j 的个数。

Input

第一行一个正整数 T ($1 \leq T \leq 10^4$)，表示数据组数。

对于每组数据，第一行一个整数 n ($2 \leq n \leq 10^6$)，表示序列长度。

第二行一个长度为 n 仅由 0, 1, ? 构成的字符串 s 表示序列，其中第 i 个字符 s_i 为 ? 表示序列中 a_i 未知待填充，否则表示 a_i 。

保证单个测试点内每组数据中 n 的和不超过 2×10^6 。

Output

对于每组数据，一行一个整数，表示最大的逆序对数。

Example

standard input	standard output
4	2
3	2
110	4
3	8
1?0	
4	
????	
7	
1?0?0?1	

Note

对于第一组样例，110 的逆序对数为 2。

对于第二组样例，可以填充为 100 或 110，逆序对数均为 2。

对于第三组样例，可以填充为 1100，逆序对数为 4。

对于第四组样例，可以填充为 1100001 或 1101001，逆序对数均为 8。

Problem G. 丢番图

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

小 A 得到了一张丢番图。
给出长度为 n 的正整数序列 a_1, a_2, \dots, a_n 和定值 t , 请你确定一个长度为 n 的非负整数序列 x_1, x_2, \dots, x_n , 对于所有 $0 \leq j < n$, 都满足:

$$\sum_{i=1}^n a_i^j \times x_i \equiv t^j \pmod{998244353}$$

请你输出序列 x_1, x_2, \dots, x_n 。注意对于 $i = 1, 2, \dots, n$, 应该有 $0 \leq x_i < 998244353$ 。
数据保证解唯一。

Input

第一行两个正整数 n, t ($1 \leq n \leq 5 \times 10^4, 1 \leq t < 998244353$)。
第二行 n 个正整数, 依次表示 a_1, a_2, \dots, a_n ($1 \leq a_i < 998244353$)。

Output

输出一行 n 个非负整数表示 x_1, x_2, \dots, x_n 。注意对于 $i = 1, 2, \dots, n$, 应该有 $0 \leq x_i < 998244353$ 。
数据保证解唯一。

Example

standard input	standard output
2 3 1 2	998244352 2

Problem H. 胡图图

Input file: `standard input`
Output file: `standard output`
Time limit: 3 seconds
Memory limit: 256 megabytes

小 A 得到了一张胡图图。
胡图图初始在格点 (x, y) 处，每一回合他可以走到 $(x \pm 1, y \pm 1)$, $(x \pm 2, y \pm 2)$, $(x \pm 1, y \pm 2)$, $(x \pm 2, y \pm 1)$ 几个位置之一，胡图图希望走到格点 (X, Y) ，请问他至少要走多少个回合。

Input

第一行一个正整数 T ($1 \leq T \leq 10^6$)，表示数据组数。
对于每组数据，一行四个整数 x, y, X, Y ($-10^9 \leq x, y, X, Y \leq 10^9$)，表示一组询问。

Output

对于每组数据，输出一行一个整数表示答案。

Example

standard input	standard output
3	0
1 2 1 2	2
1 1 3 4	49
1 1 98 98	

Problem I. 本质不同后缀

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

小 A 有一天得到了若干个字符串，但是他并不知道这些字符串的具体含义。为了发现其中的奥秘，他决定从字符串的本质不同后缀入手，计算一些有趣的信息。

字符串的后缀是从其某个位置开始到末尾的一段。例如字符串 `abc` 的后缀包括：`abc`, `bc`, `c`。

他想知道有多少本质不同的后缀。

Input

第一行一个正整数 N ($1 \leq N \leq 3 \times 10^5$)，表示小 A 得到的字符串数量。

接下来 N 行，每行一个由小写英文字母构成的字符串 S_i ，表示小 A 得到的其中一个字符串。

保证 $\sum |S_i| \leq 3 \times 10^5$ 。

Output

输出一行一个整数，表示本质不同后缀的个数。

Example

standard input	standard output
3 fbudpefs cfhjoofs edpouftu	22

Problem J. 四川省赛

Input file: standard input
Output file: standard output
Time limit: 5 seconds
Memory limit: 256 megabytes

你要参加 SCCPC 了。
你找到了一棵 n 个点的树，树上的每个点都挂了一个英文大写字符，不妨记点 i 挂的字符是 S_i 。
你想知道这棵树上有多少条包含恰好五个点的简单路径 u, v, x, y, z ，使得 $S_u S_v S_x S_y S_z$ 按顺序写出来刚好是字符串 SCCPC。

Input

第一行一个正整数 T ($1 \leq T \leq 10^4$)，表示数据组数。
对于每组数据，第一行一个整数 n ($1 \leq n \leq 10^6$)，表示树的点数。
第二行一个长度为 n 的仅由大写英文字母构成的字符串 S ，字符串的第 i 个字符 S_i 即树上第 i 个点挂的字符。
接下来 $n - 1$ 行，每行两个整数 x_i, y_i ($1 \leq x_i, y_i \leq n, x_i \neq y_i$)，表示树上有一条连接点 x_i 和 y_i 的边。
保证单个测试点内每组数据中 n 的和不超过 2×10^6 。

Output

对于每组数据，一行一个整数表示简单路径的数量。

Example

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

Problem K. 点分治

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

对于一棵 n 个点的无根树 T ，点的编号为 $1, 2, \dots, n$ ，小 A 将其按照排列 p_1, p_2, \dots, p_n 操作以如下方式得到一棵有根树 T' ：

- 1. 找到无根树 T 中在排列 p_1, p_2, \dots, p_n 中出现位置最早的点 x 。
- 2. 将 x 从 T 中删除，并往 T' 中加入 x 作为 T' 的根。
- 3. T 中剩下若干个连通块 T_1, T_2, \dots, T_k （可能 $k = 0$ ），每个连通块 T_i 仍然是一棵无根树，对每棵无根树 T_i 操作得到有根树 T'_i 。
- 4. 将每棵有根树 T'_i 加入 T' ，并将 T'_i 的根的父亲设为 x 。

现在给出一棵树 T 和操作排列 p_1, p_2, \dots, p_n ，小 A 希望得到 T 按照排列 p_1, p_2, \dots, p_n 操作后得到的有根树 T' 上每个点的父亲。

Input

第一行一个正整数 T ($1 \leq T \leq 10^4$)，表示数据组数。
对于每组数据，第一行一个整数 n ($1 \leq n \leq 10^5$) 表示树的点数。
第二行 n 个整数 p_1, p_2, \dots, p_n ($1 \leq p_i \leq n, \forall i \neq j, p_i \neq p_j$)，表示排列。
接下来 $n - 1$ 行，每行两个整数 x, y ($1 \leq x, y \leq n, x \neq y$)，表示树上的一条边。
保证单个测试点内每组数据中 n 的和不超过 10^6 。

Output

对于每组数据，一行 n 个整数，第 i 个整数表示操作后得到的有根树 T' 上点 i 的父亲，如果点 i 为根则点 i 的父亲编号为 0。

Example

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

Note

对于第一组样例，首先 $p_1 = 2$ ，所以 T' 的根为 2， T 分为连通块 $T_1 = \{2\}, T_2 = \{3\}$ ，于是 2, 3 在 T' 上的父亲均为 2。

对于第二组样例，首先 $p_1 = 2$ ，所以 T' 的根为 2， T 分为连通块 $T_1 = \{1, 3\}, T_2 = \{4\}, T_3 = \{5\}$ 。 T_2, T_3 都是单个点构成的树，于是 4, 5 的在 T' 上的父亲均为 2；而对于 $T_1 = \{1, 3\}$ ，由于 1 在序列 p 中的出现位置更靠前（ $p_2 = 1, p_5 = 3$ ），所以 T'_1 的根为 1，于是 1 在 T' 上的父亲为 2，3 在 T' 上的父亲为 1。

对于第三组样例，首先 $p_1 = 1$ ，所以 T' 的根为 1， T 分为连通块 $T_1 = \{2, 4, 5\}, T_2 = \{3\}$ 。 T_2 是单个点构成的树，于是 3 在 T' 上的父亲为 1；而对于 $T_1 = \{2, 4, 5\}$ ，由于 2 在序列 p 中的出现位置更靠前（ $p_2 = 2, p_4 = 4, p_5 = 5$ ），所以 T'_1 的根为 2，于是 2 在 T' 上的父亲为 1。继续拆分 4, 5 分别构成单独子树，其在 T' 上的父亲均为 2。

Problem L. abc

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

小 A 有一天对 Atcoder Beginner Contest 非常感兴趣，于是就打算出一道只有 abc 的题目。
给定一个只包含 a, b, c，且长度为 N 的串 S 。
定义一个字符串 S 的价值 $\text{val}(S)$ ，是 S 中出现次数最多的字符的出现次数减掉 S 中出现次数最少的字符的出现次数。注意， S 中出现次数最少的字符的出现次数不为 0，即 S 中出现次数最少的字符要在 S 中出现过。
例如 aaa 的价值为 0，aab 的价值为 1，abccc 的价值为 2。
他想知道：

$$\sum_{i=1}^N \sum_{j=i}^N \text{val}(S[i, j])$$

其中，子串 $S[i, j]$ 表示字符串 S 的第 i 个字符到第 j 个字符组成的字符串。
小 A 现在想去打 Atcoder Beginner Contest 了，所以他请你帮他回答这个问题。

Input

第一行一个 N ($1 \leq N \leq 2 \times 10^5$)，表示字符串长度。
第二行一个长度为 N 的字符串 S ，保证只包含字符 a, b, c。

Output

输出一行一个整数，表示答案。

Example

standard input	standard output
5 baaca	8

Note

$S[1, 3]$ 的价值是 1， $S[1, 4]$ 的价值是 1， $S[1, 5]$ 的价值是 2， $S[2, 4]$ 的价值是 1， $S[2, 5]$ 的价值是 2， $S[3, 5]$ 的价值是 1，所以价值和为 8。