文件名 jamo.cpp/in/out attack.cpp/in/out lie.cpp/in/out tree.cpp/in/out

题目名称	字母简记	攻与防	四月是你的谎言	嘤嘤的珂朵莉树
题目类型	传统型	传统型	传统型	传统型
每个测试点时限	C/C++ 1 秒，其他语言 2 秒	C/C++ 1 秒，其他语言 2 秒	C/C++ 1 秒，其他语言 2 秒	C/C++ 1 秒，其他语言 2 秒
内存限制	C/C++ 256MB，其他语言 512MB	C/C++ 256MB，其他语言 512MB	C/C++ 256MB，其他语言 512MB	C/C++ 256MB，其他语言 512MB
子任务数目	10	10	10	10
测试点是否等分	是	是	是	是

1. 字母简记

【题目描述】

鸡尾酒讨厌喜欢在 QQ 群里复读别人说话的人（除清楚姐姐），这种人通常被称为复读机。如果鸡尾酒是群主，那么他就会让这种人把之前群里发的所有消息全部复读一遍！（备注：复读指的是将一句话重复多遍）

现在给出一个长度为 n 的仅包含小写字母和数字的字符串，字母表示已经发的消息，而数字则表示复读机需要复读的次数。例如 $jwjnb3$ 表示将 $jwjnb$ 复读 3 遍，即 $jwjnbjwjnbjwjnb$。然而，复读并没有这么简单，鸡尾酒想让他们复读更多的次数！于是这个字符串中可能包含多个数字，当多次出现数字时，例如 $a5b2$，我们从左到右解析这个字符串，$a5$ 表示将 a 复读 5 遍，即原字符串变为 $aaaaab2$，然后遇到数字 2，再将所有消息全部复读 2 遍，即 $aaaaabaaaaab$。字符串中不会出现单独的 0，即鸡尾酒不会让他们将某一段话复读 0 遍——这根本不算惩罚！

复读机一看要复读这么多内容，瞬间被吓哭，于是找到你来帮他复读！

【输入格式】

第一行输入一个数字 t，表示本题共有 t 组数据。

接下来 t 行每行包含一个数字 n 和一个字符串 s。

【输出格式】

输出共 t 行。每行一个字符串表示需要复读的内容。

【样例 1 输入】

2

4 a2b1

5 z11y2

【样例 1 输出】

aab

zzzzzzzzzzzyzzzzzzzzzzy

【数据范围】

对于 30% 的数据，所有数字独立出现，即数字最多只有一位。

对于 100% 的数据，有 $1 \leq t \leq 100$，所有数据的字符串长度及最终原话长度小于之和等于 $5 * 10^5$

2. 攻与防

【题目描述】

有 n 个战士站成一排，分别编号为 $1,2,3\ldots n$，战士的战斗力等于他的编号 ，有一些战士只会进攻，有一些战士只会防守。现在我们要将他们从某个点开始分成两个阵营，假设这个点为 $pos(0 \le pos \le n)$，则编号小于等于 pos的战士为第一个阵营，编号大于 pos 的战士为第二个阵营。我们令第一个阵营为进攻方，第二个阵营为防守方，假设第一个阵营中能够进攻的战士战斗力总和为 w，第二个阵营中能够防守的战士战斗力总和为 v，我们希望 $|w-v|$ 的值最小，其中 || 为绝对值符号。求 $|w-v|$ 的最小值。

【输入格式】

第一行输入一个正整数 n，表示战士的数量（即后续给出的字符串的长度）。

第二行给出一个字符串 s，仅由字符 0 或 1 组成，字符串中的每一位分别代表每一位战士的属性，0 代表这个战士只会进攻，1 代表这个战士只会防守。

【输出格式】

输出一行一个整数表示答案。

【样例 1 输入】

4

0011

【样例 1 输出】

1

【样例 1 说明】

假设我们在第二个位置切割，这样左边的字符串为 0，右边的字符串为 1，代表左边有 2 个会进攻的战士，战斗力之和为 $1+2=3$，右边有 2 个会防守的战士，战斗力之和为 $3+4=7$，即 $|w-v|=|3-7|=4$。但如果我们在第三个位置切割，左边的字符串为 001，右边的字符串为 1，此时左边会攻击的战士战斗力之和还是 3，右边会防守的战士战斗力为 4，此时差值为 1，差值最小。

【样例 2 输入】

7

1000101

【样例 2 输出】

2

【样例 2 说明】

第二个样例可以在第五个位置分割，即左边的字符串为 10001，右边的字符串为 01，代表左边有 3 个会进攻的战士，2 个会防守的战士，所有会攻击战士的战斗力之和即 $w=2+3+4=9$；右边有 1 个会防守的战士，1 个会进攻的战士，所有会防守的战士的战斗力之和为 $v=7$。所以 $|9-7|=2$。

【数据范围】

对于 20% 的数据，$n\leq 10$

对于 40% 的数据，$n\leq 1000$

对于 100% 的数据，$n\leq 100000$

3. 四月是你的谎言

【题目描述】

嘤嘤最近正在看《四月是你的谎言》，看完后她觉得——呜~，太好哭了吧!。

嘤嘤$PTSD$了，现在她一看到某些单词就会嘤嘤嘤，现在有一个字符串里面包含了很多会让嘤嘤嘤嘤嘤的单词，而魔法少女$qcjj$为了让嘤嘤看到这个字符串时不会嘤嘤嘤，她决定使用魔法将字符串中的某些字符变成$'*'$。

为了尽可能使字符串保持原来的特征，魔法少女$qcjj$希望字符串改变的次数尽可能少，即$'*'$的数量尽可能少，她想让你帮他找到一个这样的方案。如果有多种方案，输出任意方案均有效。

【输入格式】

第一行输入一个整数 T 表示测试用例数。

对每一个测试用例，第一行输入一个整数 n 表示嘤嘤看到会嘤嘤嘤的单词的数量。

接下来 n 行，每行输入一个字符串 s 表示嘤嘤看到会嘤嘤嘤的单词。

接下来 1 行，输入牛牛的字符串 str 。

【输出格式】

输出一个字符串，使得字符串不包含嘤嘤看到后会嘤嘤嘤的单词，并且需要保证

字符串中′*′的数量最少，如果存在多种方案，输出任意一个均有效。

【样例 1 输入】

5

1

a

aa

1

ab

aba

1

a

bbb

2

abcd

efg

abcdefg

2

lie

april

yourlieinapril

【样例 1 输出】

**

*ba

bbb

bcdef

yourl*ein*pril

【样例 1 说明】

第一个样例，要使aa中不包含a，必须将所有字符变成*。

第二个样例，$*ba$中不包含ab，同理$a*a$中也不存在ab，所以$a*a$也是一种合法方案。

第三个样例，bbb中已经不包含a了，所以不需要改变字符。

【数据范围】

对于20%的数据，$n=1$。

对于40%的数据，str 总长度不超过1500。

对于100%的数据，$1 \le T \le 10^5$,$1 \le n \le 10$，每个单词长度不超过10，str总长度不超过$5*10^5$

4. 嘤嘤的珂朵莉树

【题目描述】

现在，嘤嘤有一棵含有 n 个结点且根为 1 号结点的树，嘤嘤将其命名为珂朵莉树，每个结点具有一个权值，每个结点与根的距离就是该结点的深度。

然后，嘤嘤会在这棵树上增加 m 条辅边，任意一条辅边不会和原来的树边或其他辅边重合（增加的辅边不会影响结点的深度）。

之后，嘤嘤会进行进行 q 次操作。

操作一：让结点 u 的权值增加 k ，并对与结点 u 相邻的结点中，深度比结点 u 小的结点重复操作一。

操作二：让结点 u 的权值增加 k ，并对与结点 u 相邻的结点中，深度比结点 u 大的结点重复操作二。

嘤嘤想知道，经过 q 次操作后，所有结点的权值分别是多少。

【输入格式】

第一行三个整数n, m, q ，分别表示树的结点个数，辅边条数，操作次数。

第二行 n 个整数 a_i，表示每个结点的初始权值。

接下来 $n-1$ 行，每行两个整数 $u,v(1 \le u,v \le n,u \ne v)$ 表示树的结构。

接下来 m 行，每行两个整数 $u,v(1 \le u,v \le n,u \ne v)$ 表示辅边连接的两个结点。

接下来 q 行,每行三个整数 $t(1 \le t \le 2),u(1 \le u \le n),k$,分别表示操作类型,操作的结点。

【输出格式】

一行 n 个整数，表示树上每个结点的权值，由于权值可能很大，请将权值对 10^9+7进行取模。

【样例 1 输入】

7 2 4

1 1 1 1 1 1 1

1 2

1 3

2 4

2 5

3 6

3 7

4 3

7 1

2 1 1

1 4 1

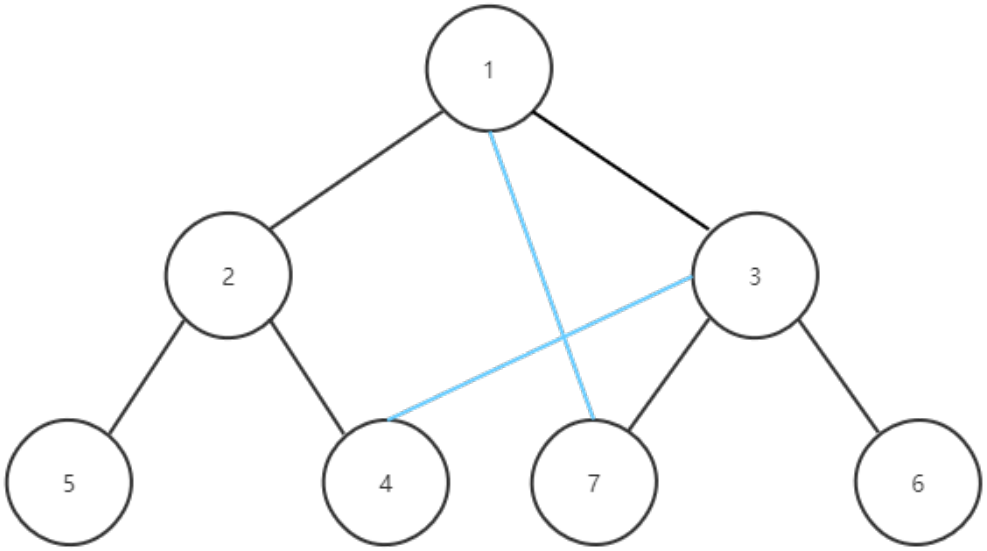
1 7 1

2 5 1

【样例 1 输出】

6 3 4 4 3 2 4

【样例 1 说明】



图中蓝色的边即为辅边。

【样例 2 输入】

6 5 3

0 0 0 0 0 0

1 2

2 3

3 4

1 5

5 6

1 3

5 3

1 6

2 6

5 4

2 1 1

2 3 1

2 6 1

【样例 2 输出】

1 1 4 5 1 4

【数据范围】

对于 20%的数据 $n \leq 1000, m = 0, q \leq 1000$， $k \leq 1000$。

对于另外 20% 的数据 $n \leq 10\,5, m = 0$,保证没有操作一。

对于 100%的数据 $n \leq 10^5$, $m \leq 2 \times 10^5, q \leq 2 \times 10^5, k \leq 10^9$ 。