

A 多国语言

时空限制：1s 256MB

文件名

language.in/language.out/language.cpp

题意

不同的国家和地区, 使用着不同的语言。

在不同的语言中, 形容猫的叫声的拟声词也有所不同, 比如中文中是“喵 (miao)", 英文中是“meow", 日文中是“にゃ- (nyaa)"。

果果想知道, 不同国家和地区的猫, 是不是真的存在不同的交流方式。为了验证这个事情, 他将 m 种语言编号为 $1, 2, \dots, m$ 。然后他随机的找

来了一只猫, 并对着猫发出 n 次猫叫声, 其中第 i 次猫叫声是第 a_i ($1 \leq a_i \leq m$) 种语言中猫叫的拟声词。

每次发出猫叫声后, 他会记录猫是否对其回应。如果猫在第 i 次回应了果果, 果果则记录 $b_i = 1$, 否则记录 $b_i = 0$ 。如果猫只对某种语言的叫声做出回应, 并对该种语言的每次叫声都做出回应, 那么果果就判断, 不同国家和地区的猫确实存在不同的交流方式, 并且认为这只猫来自该种语言的国家。此时输出该种语言的编号。

否则, 如果猫对 ≥ 2 种语言的叫声做出了回应, 或者对 1 种语言做出过回应, 但没有对每次该语言的叫声做出回应, 那么果果认为猫真可爱, 并且给它一条小鱼干。此时输出 " \wedge "。

当然也有可能, 猫根本就不回应果果的任意一次叫声, 这时果果将会很沮丧。此时输出 " $>-<$ "。

输入格式

第一行, 一个正整数 T , 表示单组测试的数据组数。

接下来每三行代表一组数据, 对于每一组数据,

其第一行, 两个正整数 n, m , 以空格相隔;

其第二行, n 个正整数 a_1, a_2, \dots, a_n , 以空格相隔, 表示 n 次猫叫的语言编号;

其第三行, n 个正整数 b_1, b_2, \dots, b_n , 以空格相隔, 表示 n 次的回应情况。

输出格式

输出 T 行, 第 i 行输出第 i 个样例的结果。

按照题意, 结果有三种可能, 分别为一个代表相应的语言编号的整数, 字符串 " \wedge " 或 " $>-<$ " (不含引号)。

输入样例

```
4
5 4
1 2 3 4 1
1 0 0 0 1
5 5
1 2 3 4 5
1 1 0 0 1
5 4
1 1 1 1 3
1 0 1 0 0
7 3
1 1 2 2 3 3 3
0 0 0 0 0 0 0
```

输出样例

```
1
^v^
^v^
>-<
```

样例解释

第一组数据，猫对每次语言1的猫叫都做出了回应，因此输出1。

第二组数据，猫对语言1,2,5都做出了回应，因此输出 ^v^。

第三组数据，猫对语言1做出了回应，但没有每次都做出回应，因此输出 ^v^。

第四组数据，猫没有做出任何一个回应，因此输出 >-<。

数据范围

对于 30% 的数据, 满足 $1 \leq n, m \leq 10$

对于 60% 的数据 满足 $1 \leq n, m \leq 10^3$

对于 100% 的数据, 满足 $1 \leq T \leq 10, 1 \leq n, m \leq 10^5, 1 \leq a_i \leq m, 0 \leq b_i \leq 1$

B double u

时空限制：1s 256MB

文件名

double.in/double.out/double.cpp

题意

手写体的 "uu" 和 "w", "nn" 和 "m" 简直让人难以辨认。某人拿到了一个手写体字符串 s , 他只能按照这个字符串某种可能的形式将其记录下来。具体来说, 对于字符串内的一个子串 "uu", 他有可能记录成 "w", 对于子串 "w", 可能记录成 "uu"。同样对于子串 "nn", 可能记录成 "m", 对于子串 "m", 可能记录成 "nn"。而对于 u, w, n, m 以外的其它字符, 则不会出现记录错误。更糟糕的是, 他记录完后, 下一个人按照他的记录进行再一次记录时又会产生同样的错误。现在拿到果果拿到了不知道被反复记录了多少次的字符串 t , 万幸的是他知道了这个字符串原本的长度为 n , 即 $|s| = n$ 。他现在希望你帮她恢复出这个字符串 s , 如果有多种可能性, 任意一种符合题意的 s 都会被认为是正确的。

输入格式

第一行一个整数 T , 表示数据组数。接下来每两行代表一组数据。对于每一组数据, 其第一行, 一个正整数 n , 表示原串 s 的长度; 其第二行, 一个字符串 t , 表示 s 经过反复记录后的字符串。保证 t 串仅由小写字母组成。

输出格式

输出 T 行, 第 i 行对应第 i 组数据的原串 s 。如果有多种可能性, 任意一种符合题意的 s 都会被认为是正确的。数据保证存在至少一个答案。

输入样例

```
3
5
abcw
7
xuwuxnmnx
3
wm
```

输出样例

```
abcuu
xwxmnm
uum
```

样例解释

第一组数据, "abcuu" 被记录成 "abcw"。第二组数据, "xwxmnm" 被记录成 "xuuuuxnnnnx", 再次被记录成 "xuwuxnmnx"。第三组数据, "uum" 被记录成 "wm", 还有另一个可能的答案是 "wnn"。

数据范围

- 对于 20% 的数据, 满足 $1 \leq n, |t| \leq 10$
- 对于 50% 的数据, 满足 $1 \leq n, |t| \leq 1000$
- 对于另外 10% 的数据, 满足字符串 t 中不含有 u, w, n, m

对于 100% 的数据, 满足 $1 \leq n, |t| \leq 100000, 1 \leq T \leq 10$ 。

所有数据保证 t 中仅含有小写字母, 保证至少存在一个答案。

C 活动

时空限制: 1s 256MB

文件名

activity.in/activityout/activity.cpp

题意

超市里, 有 n 种物品, 第 i 种物品的重量为 i , 且每种物品都有无限多个。

有一天, 超市举办了一个活动。活动的规则是: 首先你从盒子里抽取一个数字 x , 你有一个空篮子, 初始时篮子内物品总重量为 0。你每次可以将一个物品放入篮子, 但假设当前篮子内物品总重量为 w , 则只能选择一个重量 $\geq w$ 的物品放进篮子。你可以按照规则放多次, 但不能把已经在篮子里的物品取出篮子。如果你可以按照规则, 使得篮子内物品的总重量刚好为 x , 则超市就赠送你一个小礼品。

果果发现了这个活动的一个必胜策略, 他参加了很多次活动, 导致超市面临亏损。于是在果果打算再次参与活动时, 超市老板对 果果提出挑战, 如果果果可以不使用第 y 种物品的前提下, 仍然使篮子内物品的总重量刚好为 x , 则超市老板就赠送果果一份大 礼品, 否则果果将再也不许参加活动。

果果欣然接受, 他只是觉得无聊。于是他打算考考你, 在不使用第 y 种物品的前提下, 他可以有多少种合法的放物品方案, 使得 篮子内物品的总重量刚好为 x ?

两种方案不同当且仅当果果在两种方案中篮子内物品的数量不同, 或者两种方案中第 k 次放到篮子的物品种类不同。由于方案数可能很大, 他只需要你输出方案数除以 998244353 的余数就可以啦。

输入格式

第一行, 三个正整数 n, x, y , 以空格相隔。

输出格式

一行, 一个整数, 表示方案数除以 998244353 的余数。

输入样例1

5 7 3

输出样例1

3

样例解释1

3种方案分别是：

(1) 放入1，放入1，放入5

(2) 放入1，放入2，放入4

(3) 放入2，放入5

输入样例2

```
8 15 4
```

输出样例2

```
4
```

输入样例3

```
12345 4321 24
```

输出样例3

```
795333476
```

数据范围

对于 10% 的数据, 满足 $1 \leq n, x \leq 10$

对于另外20% 的数据, 满足 $1 \leq n, x \leq 1000$

对于 100% 的数据, 满足 $1 \leq n, x \leq 100000, 1 \leq y \leq n$

D 灯笼

时空限制：1s 256MB

文件名

lantern.in/lantern.out/lantern.cpp

题意

国庆长假, 果果带着何老板一起去逛街, 他们来到一个街道, 街道上从左至右悬挂了 N 盏五颜六色的灯笼。果果想要带何老板去这个街道中的一小段街区看灯笼, 具体来讲, 果果会先选择街道中的两个端点 $(u, v), u, v \in [1, N]$, 然后他们从街道从左往右数的第 u 个灯笼看到从左往右数的第 v 个灯笼。

何老板对于灯笼的喜好不同, 她给这 N 盏灯笼都给出了一个喜爱度, 第 i 盏灯笼的喜爱度为 $like_i$ 。何老板觉得好不容易出来玩, 如果逛的灯笼都不太喜欢, 甚至讨厌, 就很难受。具体来讲, 如果他们所逛的这一小段街区中所有灯笼的喜爱度之和小于 X , 何老板就不能接受。

果果不希望看到灯笼的种类数多于 M , 因为这样他会看的眼花。对于第 i 盏灯笼和第 j 盏灯笼, 如果何老板给出的喜爱度 $like_i = like_j$, 我们就认为第 i 盏灯笼和第 j 盏灯笼是同一种灯笼。

现在果果想要知道, 街道中有多少种选择街区的方式可以满足他们两个人的条件?

输入格式

第一行输入三个整数 N, M, X 。

接下来一行输入 N 个整数 $like_i$, 表示每盏灯笼的喜爱度。

输出格式

仅一个整数, 表示果果选择街区的方案数。

输入样例1

```
5 5 5
3 2 -4 2 3
```

输出样例1

```
6
```

样例解释1

合法的逛街方案可以是 $(1, 2), (2, 1), (4, 5)(5, 4), (1, 5), (5, 1)$, 一共 6 种。

输入样例2

```
5 4 -10000000000
1 2 3 4 5
```

输出样例2

23

样例解释2

排除法, 总共 5×5 种方案, 除了 $(1, 5), (5, 1)$ 出现了 5 种灯笼不满足条件, 其他都是合法的。所以答案为 $5 \times 5 - 2 = 23$ 。

数据范围

对于 30% 的测试数据, 保证 $1 \leq N \leq 10^3$ 。

对于不属于前 30% 的另 10% 的测试数据, 保证 $M = N$ 。

对于不属于前 30% 的另 10% 的测试数据, 保证 $X = -10^9$ 。

对于不属于前 30% 的另 10% 的测试数据, 保证 $like_i \geq 0$ 。

对于 100% 的测试数据, 保证 $1 \leq N \leq 10^5, 1 \leq M \leq 10^5, -10^9 \leq X \leq 10^9, -10^4 \leq like_i \leq 10^4$ 。