

Whalica Cup (Round 1) (Statement)

A. 一道题教你学会猫娘语喵

Statement

Whalica 是一只不太聪明的可爱猫娘喵 ~

这一天喵, Whalica 收到了好多好多封信喵, 每一封信上面都写了好多好多的英文字母喵, Whalica 太笨了喵, 不会读里面的内容喵, 想请你来读出这些信的内容喵 ~ 为了让猫娘 Whalica 能听得懂你在说什么喵, 你需要学会猫娘语喵 ~

具体来说喵, 如果读到了字母 " n " 喵, 猫娘们不会读出这个字母喵, 而是会本能反应地 **小喵一声** " nya " 喵; 如果读到了字母 " N " 喵, 猫娘们也不会读出这个字母喵, 而是会接着 **大喵一声** " NYA " 喵, 而对于其他字母的话喵, 只需要正常地读出来就可以了喵, 恭喜你学会猫娘语喵 ~

Whalica 已经等不及要听听信里写了什么喵, 请你帮 Whalica 读出来喵 ~ 如果还没有听懂的话喵, 可以看看下面的样例喵 ~

Input Format

第一行包含一个整数 t ($1 \leq t \leq 1000$) 喵, 表示总共有 t 封信要读喵;

对于每封信喵, 内容是一个字符串 s ($1 \leq |s| \leq 100$) 喵, 表示你要读的信的内容喵;

由于猫娘们只会读英文字母喵, 保证信件里的内容只包含小写英文字母和大写英文字母喵。

Output Format

输出共包含 t 行喵, 表示每封信件的内容读成猫娘语的形式喵。

Example

input

```
3
whalicanaya
suki
aNon
```

output

```
whalicanayaya
suki
aNYAonya
```

Note

对于样例的解释喵:

对于第一组样例喵, 信的内容里有一个小写字母 " n " 喵, 读的时候要 **小喵一声** 喵;

对于第二组样例喵, 信里没有字母 " n " 和 " N " 喵, 正常读就可以了喵;

对于第三组样例喵, 信里字母 " N " 和 " n " 先后各出现了一次喵, 读的时候要 **先大喵一声** 喵, 再 **小喵一声** 喵。

B. 你知道的, Arcaea 是一款, 后面忘了

Background

注意: 此部分背景知识只是对于游戏内容的介绍, 保证和下面题目的答案的计算无关。

Arcaea 是一款结合 3D 立体轨道与双键判定的移动端音乐游戏。其中, *Note* 是游戏中需要触击的音符, 谱面是 *Note* 的排列组合, 决定了歌曲的演奏顺序、节奏与难度。

Statement

Arcaea 真是太好玩啦, Whalica 如是说道。

在 *Arcaea* 里, 点击 *Note* 可能产生四种判定结果: "*PURE*" (大 *P*) , "*pure*" (小 *p*) , "*far*", "*lost*", 每种判定结果都对应着不同的分数, 但是 *Arcaea* 大佬一般只关心大 *P* 和小 *p* 的出现次数, 因为如果一张谱面里所有的 *Note* 的点击判定都是大 *P*, 那么就算达到了这张谱面的 **理论值**。

大 *P* 的判定区间是 $\pm 25ms$, 而小 *p* 则是 $\pm 50ms$, 也就是说, 对于一张有 n 个 *Note* 的谱面, 如果对于第 i 个 *Note*, 点击判定 $a_i \in [-25, 25]$, 那么这个 *Note* 会被判定为大 *P*, 否则, 如果 $a_i \in [-50, 50]$, 那么这个 *Note* 会被判定为小 *p*。

现在有 t 位大佬的游玩数据, 你需要判断每位大佬是否达到了理论值。

Input Format

第一行包含一个整数 t ($1 \leq t \leq 1000$), 表示下面有 t 位大佬的游玩数据;

对于每位大佬的游玩数据, 第一行一个整数 n ($1 \leq n \leq 1000$), 表示游玩的谱面总共有 n 个 *Note*;

第二行 n 个整数 a_i ($-100 \leq a_i \leq 100$), 表示每个 *Note* 的点击判定。

Output Format

输出共包含 t 行, 表示每位大佬是否达到了理论值;

如果达到理论值, 输出 *YES*, 否则输出 *NO*。

请注意, 答案的输出区分大小写。

Example

input

```
3
2
0 10
3
1 40 -20
1
-100
```

output

YES
NO
NO

Note

对于部分样例的解释：

对于第一组样例，这位大佬所有 *Note* 的点击判定都在 $[-25, 25]$ 以内，所以他达到了理论值；

对于第二组样例，第二个 *Note* 的点击判定在 $[-25, 25]$ 之外，不满足所有的 *Note* 都是大 *P*，所以他并没有达到理论值。

C. 出个区间最值操作吓似你

Statement

Mutsumi 和 Mortis 正在玩一个简单的游戏。

游戏初始给定了一个长度为 n 的数组 a 和一个整数 k ($1 \leq k \leq n \leq 10^5, 1 \leq a_i \leq 10^9$)，Mutsumi 和 Mortis 轮流进行各自的操作，Mutsumi 先手。

对于每轮游戏，Mutsumi 的每次操作会选取数组 a 中一段长度 **恰好** 为 k 的区间，将这段区间内的所有元素赋值为这个区间的 **最大值**；而 Mortis 的每次操作会选取数组 a 中一段长度 **恰好** 为 k 的区间，将这段区间内的所有元素赋值为这个区间的 **最小值**。当数组 a 的所有元素均相等时，游戏结束。

Mutsumi 和 Mortis 都单纯地渴望着一个自己想要的数值，也就是说：

尽管游戏可能永远都不会结束，Mutsumi 也还是想要不计一切代价最大化最终元素的值，而 Mortis 也会不计一切代价想要最小化最终元素的值，两人都会以 **最优方式** 进行这个游戏。

你需要计算游戏结束时数组 a 的所有元素都等于的值，或报告游戏永远不会结束。

Input Format

第一行包含一个整数 t ($1 \leq t \leq 10^4$)，表示测试数据的组数；

对于每组测试数据，第一行两个整数 n, k ($1 \leq k \leq n \leq 10^5$)，含义如题目所述；

第二行 n 个整数 a_i ($1 \leq a_i \leq 10^9$)，表示数组 a ；

数据保证单个测试点的 n 的和不超过 10^5 。

Output Format

输出共包含 t 行，表示每组测试数据的答案；

对于每组测试数据，若游戏永远不会结束，输出 -1 ，否则输出游戏结束时，数组 a 所有元素等于的值。

Example

input

```
4
1 1
5
2 1
2 3
2 2
3 6
5 3
4 6 2 3 3
```

output

```
5
-1
6
-1
```

Note

对于部分样例的解释：

对于第一组样例，数组的所有元素已经全部相等了，最终相等的元素等于 5；

对于第二组样例，Mutsumi 和 Mortis 每次操作都只能操作一个位置，所以数组元素的值永远不会改变，游戏会一直进行下去。

D. 是不是什么题加个 kth 都会变难一点

Statement

在数学中，**MEX** (Minimum Excluded) 是指一个序列中未出现的最小非负整数。但是 Whalica 觉得只是求这个也太没意思了，Whalica 开动脑筋，不小心在前面加了个 kth，这好像很喵！但是 Whalica 不会求，Whalica 很难过，Whalica 拉拉你的衣角，看起来快要掉小珍珠了，你能帮她吗？

简单来说，给定一个数组 a ，你需要求出这个数组的 **kth-MEX**，这里 **kth-MEX** 被定义为：一个序列未出现的非负整数中从小到大的第 k 个。例如，对于一个序列 $[0, 1, 4]$ ， $k = 2$ 时，它的 kth-MEX 为 3，因为 3 是这个序列未出现的非负整数中从小到大的第 2 个； $k = 5$ 时，它的 kth-MEX 为 7，因为 7 是这个序列未出现的非负整数中从小到大的第 5 个。

如果你能帮 Whalica 解决这题的话，她会好好报答你的。

Input Format

第一行包含一个整数 t ($1 \leq t \leq 10^4$)，表示测试数据的组数；

对于每组测试数据，第一行两个整数 n, k ($1 \leq n \leq 10^5, 1 \leq k \leq 10^{18}$)，含义如题目所述；

第二行 n 个整数 a_i ($0 \leq a_i \leq n$)，表示数组 a ；

数据保证单个测试点的 n 的和不超过 10^5 。

Output Format

输出共包含 t 行，表示每组测试数据的答案。

Example

input

```
3
2 2
0 1
3 1
1 3 2
3 4
0 2 3
```

output

```
3
0
6
```

E. 质数筛太没意思了所以我们加入了合数筛

Statement

合数 是指除了 1 和其本身之外还有其他正因子的正整数。

Whalica 做了很多质数筛的题，觉得质数筛实在是太没意思了，于是她想出了一个听上去很好玩的东西：合数筛。Whalica 绞尽脑汁，想出了一个感觉会很有意思的题目。Whalica 定义函数 $f(n)$ （其中 n 是合数）表示整数 n 是从 1 到 n 的第 $f(n)$ 个合数，例如： $f(4) = 1$ ，因为 4 是从 1 到 4 的第一个合数（1 既不是质数也不是合数）； $f(8) = 3$ ，因为 8 是从 1 到 8 的第 3 个合数，1 到 8 包含的三个合数为：4, 6, 8。

Whalica 会一次性给出 q ($1 \leq q \leq 10^5$) 段区间 $[l, r]$ ($1 \leq l \leq r \leq 5 \times 10^6$)，对于每段区间，你需要计算：

$$\sum_{i=l}^r i \cdot f(i), \text{ 其中 } i \text{ 是合数}.$$

也就是对区间 $[l, r]$ 内的所有合数 i ，求 $i \cdot f(i)$ ，并对这些值求和，由于结果可能会很大，请将结果对 998244353 取模。

Input Format

第一行包含一个整数 q ($1 \leq q \leq 10^5$)，表示总共有 q 段区间；

随后共计 q 行，每行包含两个整数 l, r ($1 \leq l \leq r \leq 5 \times 10^6$)，表示区间的左右端点。

Output Format

输出共包含 q 行，表示对于每段区间，你计算的答案；

结果对 998244353 取模。

Example

input

```
4
1 3
4 8
213 1445
1 5000000
```

output

```
0
40
719846654
751614103
```

Note

对于部分样例的解释：

对于第一组样例，区间 $[1, 3]$ 内没有合数，所以答案是 0；

对于第二组样例，区间 $[4, 8]$ 共包含 3 个合数 4, 6, 8，其中 $4 \times f(4) = 4 \times 1 = 4$, $6 \times f(6) = 6 \times 2 = 12$, $8 \times f(8) = 8 \times 3 = 24$ ，所以答案是 $4 + 12 + 24 = 40$ 。

F. 怎么有人出的题自己不会做的

Statement

Whalica 讨厌计数题。Whalica 遇到计数题，有 99% 的概率不会做。她也想让别人体验一下做不出计数题的感觉，于是她也出了一道计数题：

给定一个长度为 n 的数组 a ($1 \leq n \leq 10^5, 1 \leq a_i \leq 10^9$) 和两个整数 k 与 s ($0 \leq k \leq 10^9, 0 \leq s \leq 10^{18}$)，你需要计算数组 a 中有多少个 **子数组** 满足：

1. 子数组的左右端点 l, r 满足 $1 \leq l \leq r \leq n$;
2. $a_l \geq k$;
3. $a_r \geq k$;
4. 子数组的和 $a_l + a_{l+1} + \cdots + a_{r-1} + a_r \geq s$;

这里，**子数组** 指的是数组中连续的一段，即选定两个下标 l, r 满足 $1 \leq l \leq r \leq n$ ，那么称新数组 $[a_l, a_{l+1}, \cdots, a_{r-1}, a_r]$ 为 a 的一个 **子数组**。

Whalica 心满意足地出完了题，当她准备写 *std* 的时候猛然发现，自己好像甚至不会做这题，那很坏了。于是她找到了你，Whalica 最好的朋友，你能帮帮她做出这题吗？

Input Format

第一行包含一个整数 t ($1 \leq t \leq 10^4$), 表示测试数据的组数;

对于每组测试数据, 共包含两行, 第一行包含三个整数 n, k, s ($1 \leq n \leq 10^5, 0 \leq k \leq 10^9, 1 \leq s \leq 10^{18}$), 含义如题目所述;

第二行 n 个整数 a_i ($1 \leq a_i \leq 10^9$), 表示数组 a 。

数据保证单个测试点的 n 的和不超过 10^5 。

Output Format

输出共包含 t 行, 每行表示你计算的答案。

Example

input

```
2
5 5 10
6 2 8 5 1
3 100 1000
1 2 3
```

output

```
3
0
```

Note

对于样例的解释:

对于第一组样例, 可以找到三个合法的子数组, 它们分别是 $[6, 2, 8]$, $[6, 2, 8, 5]$, $[8, 5]$;

对于第二组样例, 可以证明, 没有合法的子数组。

G. 怎么有人出的题自己不会做的 again

Statement

卧槽, Whalica 彻底怒了。Whalica 很难过, 因为自己竟然做不出自己出的题, 这也太难受了。为了锻炼自己的计数能力, Whalica 决定刻苦训练计数题, Whalica 训练了许久, 又凭借着自己的知识, 出了这么一题:

给定两个参数 n, m ($1 \leq n \leq 10^{18}, 1 \leq m \leq 2 \times 10^5$), 你需要计算有多少个长度为 n 的数组满足以下条件:

1. 对于任意 $i \in [1, n]$, 都有 $1 \leq a_i \leq m$;
2. 存在唯一的下标 p , 使得 a_p 是数组 a 的最大值;
3. 存在唯一的下标 q , 使得 a_q 是数组 a 的最小值。

Whalica 再一次心满意足地出完了题, 这次, Whalica 想要自己做出这道题! 但是现在你就是 Whalica, 你应该会做的吧? 由于最终的答案可能很大, 请将结果对 998244353 取模。

Input Format

第一行包含一个整数 t ($1 \leq t \leq 10^4$), 表示测试数据的组数;

对于每组测试数据, 一行两个整数 n, m ($1 \leq n \leq 10^{18}, 1 \leq m \leq 2 \times 10^5$), 含义如题所述;

数据保证单个测试点的 m 的和不超过 2×10^5 。

Output Format

输出共包含 t 行, 每行包含一个整数, 表示你计算好的答案, 结果对 998244353 取模。

Example

input

```
2
2 3
1 99
9 4
```

output

```
6
99
9360
```

Note

对于部分样例的解释:

对于第一组样例, 满足条件的数组有 $[1, 2], [1, 3], [2, 1], [2, 3], [3, 1], [3, 2]$, 共有 6 种不同的数组;

对于第二组样例, 满足条件的数组共有 99 种。

H. 所以什么是生成排列

Statement

Whalica 也讨厌排列题。Whalica 在清理自己的 unsolved 题目列表的时候发现, 怎么一堆排列题!

Whalica 好累, 呜呜着找到了你, 让你帮她清理一下 unsolved 题目列表, 快去帮帮她吧!

其中一题是这样的:

我们定义一个长度为 n 的排列 p 是 k -生成排列, 当且仅当这个排列 p 能够经过恰好 k 次 **对换**, 使得做完所有对换操作后的 p **严格单调递增**。

其中, 长度为 n 的排列指的是一个长度为 n 的数组, 满足整数 $1, 2, \dots, n-1, n$ 在这个数组中都 **恰好出现一次**; 记操作“选定两个下标 i, j 满足 $1 \leq i < j \leq n$, 交换 p_i 和 p_j ”为一次 **对换**。

给定一个长度为 n 的排列 p 和一个整数 k , 你需要判断排列 p 是否是一个 k -生成排列。

Input Format

第一行包含一个整数 t ($1 \leq t \leq 10^4$), 表示测试数据的组数;

对于每组测试数据, 第一行两个整数 n, k ($1 \leq n \leq 5 \times 10^5, 0 \leq k \leq 5 \times 10^5$), 含义如题目所述;

第二行 n 个整数 p_i , 表示排列 p , 保证给出的所有 p_i 形成的是一个排列。

数据保证单个测试点的 n 的和不超过 5×10^5 。

Output Format

输出共包含 t 行, 表示每组测试数据的答案;

对于每组测试数据, 如果 p 是一个 k -生成排列, 请输出 *YES*, 否则输出 *NO*。

请注意, 答案的输出区分大小写。

Example

input

```
4
1 0
1
1 1
1
3 1
3 1 2
4 6
1 2 3 4
```

output

```
YES
NO
NO
YES
```

Note

对于样例的解释:

对于第一组样例, 给定的排列的长度为 1, 初始就一定是顺序的;

对于第二组样例, 需要恰好进行一次操作, 但是无法进行 **对换**, 所以无法变成完全顺序;

对于第三组样例, 可以证明, 无论如何选择下标, 排列都不可能是顺序的;

对于第四组样例, 我们可以选定 $i = 3, j = 4$ 进行恰好 6 次对换, 能够将排列变为完全顺序。