

# 洛谷 NOIP2023 模拟赛

## 第一试

### LGR-165

时间：2023 年 11 月 11 日 08:00 ~ 12:30

题目名称	种树	汪了个汪	挑战 NPC IV	四暗刻单骑
题目类型	传统型	传统型	传统型	传统型
目录	plant	woof	npc	mahjong
可执行文件名	plant	woof	npc	mahjong
输入文件名	plant.in	woof.in	npc.in	mahjong.in
输出文件名	plant.out	woof.out	npc.out	mahjong.out
每个测试点时限	1.0 秒	1.0 秒	2.0~5.0 秒	1.0 秒
内存限制	512 MiB	512 MiB	512 MiB	512 MiB
测试点数目	25	20	25	25
测试点是否等分	否	是	是	是

提交源程序文件名

对于 C++ 语言	plant.cpp	woof.cpp	npc.cpp	mahjong.cpp
-----------	-----------	----------	---------	-------------

编译选项

对于 C++ 语言	-O2 -std=c++14 -static
-----------	------------------------

注意事项（请仔细阅读）

1. 文件名（程序名和输入输出文件名）必须使用英文小写；
2. C++ 中函数 main() 的返回值类型必须是 int，程序正常结束时的返回值必须为 0。
3. 选手源程序必须存放于题目指定的目录下。
4. 程序可使用的栈空间大小与该题内存空间限制一致。
5. 若无特殊说明，每道题的源代码大小限制为 100KiB，输入与输出中同一行的相邻整数、字符串等均使用一个空格分隔，结果比较方式为全文比较（忽略行末空格、文末回车）。
6. 统一评测时采用的机器配置为 Intel(R) Xeon(R) Platinum 8369B CPU @ 2.70GHz，内存 16 GB。上述时限以此配置为准。
7. 评测在 Linux 下进行。数据与附加样例文件均为 Linux 格式的。
8. 选手不得在源程序中使用内嵌汇编，也禁止以任何方式修改程序的编译选项。

## 种树 (plant)

### 【题目背景】

小 Rf 不是很喜欢种花，但他喜欢种树。

### 【题目描述】

路边有  $n$  棵树，每棵树的高度均为正整数，记作  $p_1, p_2 \dots p_n$ 。

定义一棵树的宽度为它高度的正因数个数，这些树能覆盖的距离为它们宽度的乘积，你想请你的朋友们来乘凉，但你发现这些树能覆盖的距离不够多。

于是你买了总量为  $w$  单位的神奇化肥。你可以施若干次肥，每次你可以使用  $k$  单位化肥（要求  $k$  必须为当前化肥量的正因数），让任意一棵树的高度乘上  $k$ ，同时你剩余的化肥量也会除以  $k$ 。每次施肥的树可任意选择，且每次施肥选择的树不需相同。

你需要最大化这些树所能覆盖的距离，并输出这个最大距离。答案对 998244353 取模。

### 【输入格式】

从文件 `plant.in` 中读入数据。

第一行，两个正整数  $n$  与  $w$ 。

第二行  $n$  个正整数  $p_1, p_2 \dots p_n$ 。

### 【输出格式】

输出到文件 `plant.out` 中。

仅一行一个整数，代表答案对 998244353 取模后的结果。

### 【样例 1 输入】

```
1 3 60
2 8 243 250
```

### 【样例 1 输出】

```
1 2304
```

**【样例 1 解释】**

- 第一次施肥，向第一棵树施 15 单位的肥，使其高度变成 120，剩余 4 单位的化肥。
- 第二次施肥，向第二棵树施 4 单位的肥，使其高度变成 972，剩余 1 单位的化肥。
- 这时候，三棵树的宽度分别为 16, 18, 8，所能覆盖的距离为 2304，为最优解。

**【样例 2】**

见选手目录下的 *plant/plant2.in* 与 *plant/plant2.ans*。

**【样例 3】**

见选手目录下的 *plant/plant3.in* 与 *plant/plant3.ans*。

**【子任务】**

测试点	$n \leq$	$p_i$	$w$	单点分值
1 ~ 5	$10^4$	$= 1$	$= 1$	1
6 ~ 10		$\leq 10^4$		$\leq 10^4$
11 ~ 15	1		6	
16 ~ 20	5			
21 ~ 25	$10^4$			

对于 100% 的数据，保证  $1 \leq n \leq 10^4$ ， $1 \leq p_i \leq 10^4$ ， $1 \leq w \leq 10^4$ 。

## 汪了个汪 (woof)

### 【题目背景】

你说得对，但是小 P 在 [NOIP2022] 喵了个喵中没有输出操作次数，获得了 0 分的好成绩。

### 【题目描述】

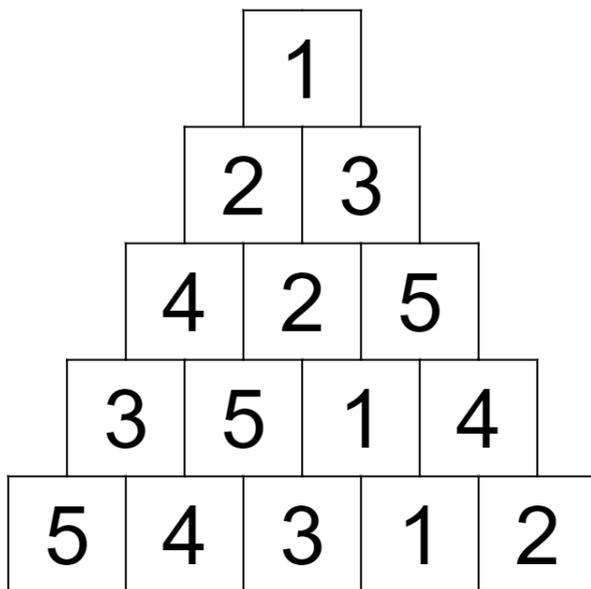
小 P 喜欢上了一款叫做《汪了个汪》的游戏。这个游戏有一个牌堆和一个金字塔形的棋盘，总共有 3 关。具体地，如图所示，棋盘的边长为  $n$ ，第  $i$  行有  $i$  个格子，共  $\frac{n(n+1)}{2}$  个格子。

牌堆中有  $1, 2 \dots n$  的数字卡片各无穷多张。你需要将这些数字卡片放到对应的棋盘格子中，每个格子恰好放一张数字卡片，要求满足棋盘的每一行的第一个元素互不相同。

小 P 发现，这个游戏的难度会随着关卡编号而增加：

- 在第 0 关中，你不必满足其他条件。
- 在第 1 关中，你需要保证一行内相邻的两个数互不相同，且所有由任意一行内相邻两个数组成的无序二元组互不相同。
- 在第 2 关中，你需要满足第 1 关的限制，并且一行内的所有数必须互不相同。

例如，下面是  $n = 5$  时可以通过第 2 关的摆放方式：



洛谷@pjytk

现在给定  $n$  与关卡编号，请你帮小 P 找出一种合适的摆放方式来通过这一关。可以证明在游戏限制下一定存在一种过关方式。

**【输入格式】**

从文件 *woof.in* 中读入数据。

仅一行，包含两个整数  $n, t$ ，其中  $t$  表示关卡编号。

**【输出格式】**

输出到文件 *woof.out* 中。

输出  $n$  行，第  $i$  行包含  $i$  个正整数（以空格分隔），表示棋盘第  $i$  行从左到右所有的数。

如果有多种合法的解，你可以输出任何一种。

**【样例 1 输入】**

```
1 2 1
```

**【样例 1 输出】**

```
1 1
2 2 1
```

**【样例 2 输入】**

```
1 5 2
```

**【样例 2 输出】**

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

## 【子任务】

测试点编号	$n \leq$	$t =$	特殊性质
1	6	0	无
2		2	
3 ~ 4	4000		
5 ~ 7	500	1	无
8 ~ 13		2	
14 ~ 16	4000	1	
17 ~ 20		2	

- 特殊性质 A: 保证  $n + 1$  或  $n + 2$  为质数。  
对于 100% 的数据, 保证  $1 \leq n \leq 4000$ ,  $t \in \{0, 1, 2\}$ 。

## 【提示】

本题下发校验器 (`checker.cpp`)。将 `checker.cpp` 编译成可执行文件 `checker` 后, 在当前目录执行 `checker woof.in woof.out woof.ans` 即可校验你的答案是否符合规范。其中 `woof.in` 可以替换为对应输入文件名称, `woof.out` 可以替换为对应输出文件名称, 也即构造结果。`woof.ans` 可以为任意文件。

返回结果说明:

- `The numbers are not in the valid range.:` 说明你的输出不满足每个数字都在  $1 \sim n$  的范围内。
- `The first column does not satisfice.:` 说明你的输出不满足每行开头的数互不相同。
- `The pairs of numbers are not distinct.:` 说明你的输出不满足所有由任意一行内相邻两个数组成的无序二元组互不相同。
- `The adjacent numbers are not distinct.:` 说明当前关卡编号  $\geq 1$  且你的输出不满足关卡 1 的条件。
- `The numbers in a row are not distinct.:` 说明当前关卡编号  $\geq 2$  且你的输出不满足关卡 2 的条件。
- `Well done.:` 说明你的构造满足要求。

## 挑战 NPC IV (npc)

### 【题目背景】

要是什么都和 NPC 问题一样简单就好了啊。

### 【题目描述】

小 R 想为小 H 写一首情诗。他现在想出了  $n$  个句子，每个句子的优美度分别为  $1, 2 \dots n$ 。小 R 需要按照一定的顺序将他们组合起来，拼成一首完整的诗。换句话说，小 R 需要重新排列这  $n$  个句子，形成一个  $1 \sim n$  的排列  $p_1, p_2 \dots p_n$ ；第  $i$  行诗句的优美度就是原先第  $p_i$  个句子的优美度，也就是  $p_i$ 。

不过，由于小 R 是位 OIer，所以他的文采并不是很稳定。他实际上会严重高估自己诗句的优美程度。若一条诗句在小 R 眼里的优美度为  $x$ ，那么小 H 认为它的优美度是  $x$  在二进制表示下最低位的 1 的位置。其中，小 H 认为最低位的位置是 1，次低位为 2，以此类推。也就是说，小 H 眼中的优美度  $f(x)$  为  $1 + \log_2 \text{lowbit}(x)$ 。

小 R 知道，小 H 拿到诗后，只会选取诗的一段来看，而她感受到的优美度是所有她读到的诗句之和。具体的，若诗有  $n$  句，则小 H 会在  $[1, n]$  的所有长度  $> 0$  的区间中抽取一个  $[l, r]$ ，感受到  $\sum_{l \leq i \leq r} f(p_i)$  的优美度。小 R 为了衡量一首诗的优美度，决定将一首诗的总优美度定义为所有情况下小 H 感受到的优美度之和。

照理来说，总优美度最大的组合方式必然是最好的。遗憾的是，在小 R 的精密计算下，他发现，只有他选择总优美度恰好为第  $k$  小的情诗时，他才最有可能和小 H 走到一起。于是，小 R 想要知道，对于  $n!$  首本质不同的诗，第  $k$  小可能的总优美度是多少。两首诗本质相同当且仅当原排列  $p_1 \dots p_n$  相同。

小 R 发现这是一个 NPC 问题，他只好来向你求助了。由于总优美度过于巨大，你只需要帮他求出答案对 998244353 取模后的结果。

特别的，小 R 写了  $q$  组诗句，所以你需要分别回答他的  $q$  个问题。

### 【输入格式】

从文件 `npc.in` 中读入数据。

第一行一个正整数  $q$ ，表示诗句的组数。

对于每组数据，仅一行两个正整数  $n, k$  描述小 R 的问题。

### 【输出格式】

输出到文件 `npc.out` 中。

对于每组诗句，输出一行一个整数，表示第  $k$  小的总优美度对 998244353 取模后的结果。

**【样例 1 输入】**

```
1 2
2 3 2
3 3 6
```

**【样例 1 输出】**

```
1 13
2 14
```

**【样例 1 解释】**

例如，当  $p = [1, 3, 2]$  时，小 H 眼中每句诗的优美度分别为  $[1, 1, 2]$ 。那么：

- 当  $l = 1, r = 1$  时，优美度之和为 1。
- 当  $l = 1, r = 2$  时，优美度之和为 1。
- 当  $l = 1, r = 3$  时，优美度之和为 2。
- 当  $l = 2, r = 2$  时，优美度之和为  $1 + 1 = 2$ 。
- 当  $l = 2, r = 3$  时，优美度之和为  $1 + 2 = 3$ 。
- 当  $l = 3, r = 3$  时，优美度之和为  $1 + 1 + 2 = 4$ 。

所以  $p = [1, 3, 2]$  的总优美度为  $1 + 1 + 2 + 2 + 3 + 4 = 13$ 。

对于所有  $3! = 6$  个排列  $p$ ，其总优美度从小到大排序后分别为 13, 13, 13, 13, 14, 14，因此当  $k = 2$  与  $k = 6$  时答案分别为 13 和 14。

**【样例 2 输入】**

```
1 5
2 4 1
3 4 10
4 4 16
5 4 20
6 4 24
```

**【样例 2 输出】**

```
1 32
2 34
```

```
3 36
4 36
5 38
```

**【样例 3 输入】**

```
1 10
2 10000000000000000000 10000000000000000000
3 1145141919810 19260817998244353
4 15 131413141314
5 36 93930322810121243
6 172 354354645654567654
7 666 233
8 1048576 2147483648
9 1000000007 1000000009
10 99824 44353
11 10 1
```

**【样例 3 输出】**

```
1 36226088
2 846277092
3 1096
4 12356
5 1239174
6 70731494
7 274614617
8 511280969
9 625722816
10 330
```

**【子任务】**

本题各测试点时间限制不相同。具体地，每个点的时间限制为  $\max(q \times 0.5, 2)$  s。

测试点编号	$n$	$k \leq$	$q =$
1 ~ 3	$\leq 10$	$n!$	2
4 ~ 8	$\leq 10^3$	2	7
9 ~ 13	$\in [10^5, 10^6]$	$\min(10^{18}, n!)$	
14 ~ 17	$\leq 10^6$		
18 ~ 25	$\leq 10^{18}$		10

对于 100% 的数据，保证  $1 \leq n \leq 10^{18}$ ， $1 \leq k \leq \min(10^{18}, n!)$ ， $1 \leq q \leq 10$ 。

## 四暗刻单骑 (mahjong)

### 【题目描述】

Alice 和 Bob 很喜欢打麻将。他们在对麻将规则熟悉后，开始对「四暗刻单骑」感兴趣。而在这局游戏中，Alice 和 Bob 都已经集齐了四暗刻，处于听牌状态并准备「四暗刻单骑」，于是我们将这样的局面简化如下：

- 一张麻将牌可以用一个范围在  $[1, k]$  内的正整数表示，数字相同的牌相同，数字不同的牌不相同。
- Alice 和 Bob 手中各有 1 张牌作为手牌。两人轮流进行摸牌，每次摸牌的玩家会得到一张牌堆顶部的牌，Alice 先进行。摸牌后会有 2 张手牌，此时需要选择一张牌打出。打出的牌双方可见。
- 当摸牌时两张手牌相同时，或当前对方打出的牌和自己目前手牌相同时，该玩家「和牌」并获胜，游戏结束。

若牌摸完后无玩家「和牌」，则判为「荒牌流局」，此时判定两位玩家平局。

现在 Alice 和 Bob 都绝顶聪明，并且已经得知了牌堆顶部的所有牌，以及对方手牌。他们都希望自己可以「和牌」并获胜，若自己无法「和牌」就会尽可能阻止对方「和牌」。

你现在拿到了  $n$  张麻将牌组成的  $a$  数组，下标依次为  $1 \dots n$ 。现在有  $q$  次询问，每次会给定  $x, y, l, r$  表示：若目前 Alice 手牌为  $x$ ，Bob 手牌为  $y$ ，且按顺序取出  $a$  中下标为  $[l, r]$  的所有牌作为游戏牌堆，其中牌  $a_l$  位于牌堆顶部，Alice 和 Bob 按要求进行游戏，最后结局如何。

询问之间相互独立。特别地，保证  $l$  为奇数。

### 【输入格式】

从文件 *mahjong.in* 中读入数据。

第一行三个正整数  $n, m, k$ 。

接下来一行  $n$  个正整数，依次表示  $a_1, a_2 \dots a_n$ 。

接下来  $m$  行，每行四个正整数  $x, y, l, r$ ，表示一次询问。

### 【输出格式】

输出到文件 *mahjong.out* 中。

对于每次询问，输出一行一个字符：如果 Alice 获胜，输出 **A**；如果 Bob 获胜，输出 **B**；如果平局，输出 **D**。

### 【样例 1 输入】

```
1 12 3 5
2 2 3 1 2 3 4 1 3 1 5 4 3
3 1 2 5 6
4 5 5 7 12
5 3 4 3 7
```

**【样例 1 输出】**

```
1 D
2 B
3 A
```

**【样例 1 解释】**

在第 1 组询问中，牌堆自顶至底依次是 3,4，Alice 手牌为 1，Bob 手牌为 2。不难发现此局面会导致「荒牌流局」。

在第 2 组询问中，牌堆自顶至底依次是 1,3,1,5,4,3，Alice 手牌为 5，Bob 手牌为 5。此时 Bob 只需要一直保留这张 5，就可以在摸上下一张 5 时「和牌」；而 Alice 不能打出 5，因为一旦打出就会导致 Bob 立刻「和牌」。

在第 3 组询问中，牌堆自顶至底依次是 1,2,3,4,1，Alice 手牌为 3，Bob 手牌为 4。Alice 第一局摸上一张 1，她打出这张 1。Bob 第一局摸上一张 2，他无论是否打出这张 2，Alice 都可以在下回合「和牌」。

**【样例 2 输入】**

```
1 7 6 3
2 2 3 3 3 1 3 3
3 1 2 5 7
4 1 1 5 6
5 1 3 1 6
6 2 3 7 7
7 1 3 3 5
8 1 2 1 4
```

**【样例 2 输出】**

1 A  
2 A  
3 B  
4 D  
5 B  
6 D

### 【样例 3】

见选手目录下的 *mahjong/mahjong3.in* 与 *mahjong/mahjong3.ans*。

### 【样例 4】

见选手目录下的 *mahjong/mahjong4.in* 与 *mahjong/mahjong4.ans*。

### 【子任务】

测试点编号	$n \leq$	$m \leq$	$k \leq$	特殊性质
1	3	3	3	A, B
2	5	5	5	无
3 ~ 5	100	100	100	
6 ~ 7	2000	2000	2000	
8 ~ 10	$5 \times 10^4$	50	$5 \times 10^4$	
11	$2 \times 10^5$	$2 \times 10^5$	2	
12			80	
13			$2 \times 10^5$	A, B
14 ~ 15	B			
16	C			
17 ~ 20	$10^5$	$10^5$	$10^5$	无
21 ~ 25	$2 \times 10^5$	$2 \times 10^5$	$2 \times 10^5$	

- 特殊性质 A: 保证每次询问  $l = 1$ 。
- 特殊性质 B: 保证每次询问  $r = n$ 。
- 特殊性质 C: 保证每次询问  $x = y$ 。

对于 100% 的数据, 保证  $3 \leq n \leq 2 \times 10^5$ ,  $1 \leq m \leq 2 \times 10^5$ ,  $1 \leq a_i, x, y \leq k \leq n$ ,  $1 \leq l \leq r \leq n$ , 保证  $l$  是奇数。