

2025 山东理工大学编程免冲刺班 Contest 5

LeafSeek

2025 年 11 月 24 日

题目名称	东之阙	下个夏天的烟火	爆燃	苦若吞沙
题目类型	传统题	传统题	传统题	传统题
目录	sequence	backpack	alpha	omega
可执行文件名	sequence	backpack	alpha	omega
输入文件名	sequence.in	backpack.in	alpha.in	omega.in
输出文件名	sequence.out	backpack.out	alpha.out	omega.out
测试点时间限制	2 秒	2 秒	2 秒	2 秒
测试点空间限制	512 MB	512 MB	512 MB	512 MB
题目总分值	100	100	100	100
子任务个数	25	13	6	10
子任务是否等分	是	否	否	是

A. 东之阙

题目描述

Sophia 在翻阅古籍时发现了一种被远古人类崇拜的序列图腾。她决定研究这种图腾，以加深对远古文化的理解。在现代看来，这些图腾可以表示为一个长度为 n 的正整数数组，每个元素不超过 k 。Sophia 需要研究的是“相似图腾”的概念。

相似图腾是指两个数组包含相同的元素（元素出现次数相同，即排序后两个数组相同）。对于两个相似图腾 a 和 b ，定义它们的序关系如下：

1. 若存在一个位置 i ，使得 $a_1 = b_1, a_2 = b_2, \dots, a_{i-1} = b_{i-1}$ 且 $a_i < b_i$ ，则认为 $a < b$ 。
2. 若存在一个位置 i ，使得 $a_1 = b_1, a_2 = b_2, \dots, a_{i-1} = b_{i-1}$ 且 $a_i > b_i$ ，则认为 $a > b$ 。
3. 否则， $a = b$ 。

该序关系构成了一个全序。

在特殊节日，古人会列出所有与原始图腾相似的图腾，并按上述序关系从小到大排列，作为祈祷的标志。Sophia 的任务是确定原始图腾在所有相似图腾中的排名。

由于相似图腾数量可能非常庞大，Sophia 只需要输出排名对 998244353 取模的结果。

输入格式

从文件 `sequence.in` 中读入数据。

输入由多行组成：

- 第一行包含一个整数 id ，表示测试点的编号。
- 第二行包含两个正整数 n 和 k ，分别表示图腾数组的长度和数组中元素的最大值。
- 第三行包含 n 个正整数 x_1, x_2, \dots, x_n ，表示原始图腾数组。

输出格式

输出到文件 `sequence.out` 中。

输出一个整数，表示原始图腾在所有相似图腾中的排名（从 1 开始），结果对 998244353 取模。

样例 1

样例 1 输入

```
0
5 3
2 1 3 2 3
```

样例 1 输出

```
8
```

样例 2

样例 2 输入

```
0
10 9
1 9 1 2 6 9 8 8 2 4
```

样例 2 输出

```
35459
```

附加样例 1 – 4

附加样例 1 – 4 输入

见选手目录下的 `sequence/ex_sequence1-4.in`。

附加样例 1 – 4 输出

见选手目录下的 `sequence/ex_sequence1-4.out`。

数据范围与说明

对于所有数据，保证 $1 \leq x_i \leq k \leq n \leq 3 \times 10^6$ 。

测试点编号	测试点分值	$n \leq$	$k \leq$	特殊性质
1	4	5	3	保证 $x = [2, 1, 3, 2, 3]$
2	4	10		保证 $x_1 \sim x_n$ 是 $1 \sim n$ 的排列
3	4	10	2	
4	4	500		
5	4	5×10^5		保证 $x_i \geq x_{i+1}$
6	4	10^4		保证 $x_1 \sim x_n$ 是 $1 \sim n$ 的排列
7	4	5×10^5		保证 $x_1 \sim x_n$ 是 $1 \sim n$ 的排列
8	4	5000	2	
9	4	10^4	2	
10	4	5×10^5	2	
11	4	10^4	100	
12	4	5×10^5	100	
13	4	10^4	1000	
14	4	5×10^4		
15	4	10^5		
16	4	2.5×10^5		保证 $x_1 \sim x_n$ 是 $1 \sim n$ 的排列
17	4	5×10^5		
18	4	7.5×10^5		
19	4	10^6		
20	4	1.25×10^6		
21	4	1.5×10^6		
22	4	1.75×10^6		
23	4	2×10^6		
24	4	2.5×10^6		
25	4	3×10^6		

B. 下个夏天的烟火

题目描述

Alice 拥有一家商店，里面有 n 件商品，编号为 $1, 2, \dots, n$ 。第 i 件商品的效益是 v_i ，其价格为 a_i 元。Bob 计划明天光顾 Alice 的商店。Bob 总是以最优方式购买商品。假设 Bob 有 t 元，他会购买一组商品，使得这些商品的总效益最大，并且总价格不超过 t 。

然而，Alice 知道 Bob 很富有，所以她决定提高一些商品的价格。具体来说，Alice 选择一对整数 l 和 r ，其中 $1 \leq l \leq r \leq n$ ，并提高所有编号在 l 和 r 之间（包括 l 和 r ）的商品的价格。当 Bob 明天来时，他需要为编号 $l \leq i \leq r$ 的第 i 件商品支付 b_i 元，而不是 a_i 元。

然而，Alice 并不知道 Bob 有多少钱，她只知道 t 是一个从范围 $[1, k]$ 中均匀随机选择的整数。

Alice 不想让 Bob 能够购买太多的好商品，因此她想知道，有多少对整数 l 和 r 可以选择，使得 Bob 购物清单的期望总效益

$$\frac{f(1) + f(2) + \dots + f(k)}{k}$$

不会超过 E ，其中 $f(t)$ 表示当 Bob 有 t 元时，购物清单的总效益。

请编写一个程序，帮助 Alice 确定有多少对整数 l 和 r 是合法的。

输入格式

从文件 `backpack.in` 中读入数据。

输入包含一个单独的测试用例。

第一行包含三个整数 n ， k 和 E ($1 \leq n, k \leq 2 \times 10^5, n \times k \leq 10^7, 1 \leq E \leq 10^9$)。

接下来的 n 行中，每行包含三个整数 v_i ， a_i 和 b_i ($1 \leq v_i \leq 10^4, 1 \leq a_i < b_i \leq k$)，表示第 i 件商品的效益、初始价格和提高后的价格。

输出格式

输出到文件 `backpack.out` 中。

输出一个整数，表示合法的整数对 l 和 r 的数量。

样例 1

样例 1 输入

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

样例 1 输出

```
1
```

样例 2

样例 2 输入

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

样例 2 输出

```
3
```

样例 3

样例 3 输入

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

样例 3 输出

```
7
```

附加样例 1 – 4

附加样例 1 – 4 输入

见选手目录下的 `backpack/ex_backpack1-4.in`。

附加样例 1 – 4 输出

见选手目录下的 `backpack/ex_backpack1-4.out`。

数据范围与说明

对于所有数据，保证 $1 \leq n, k \leq 2 \times 10^5$ ， $n \times k \leq 10^7$ ， $1 \leq E \leq 10^9$ ，且对于每个 i 有 $1 \leq v_i \leq 10^4$ ， $1 \leq a_i < b_i \leq k$ 。

子任务编号	子任务分值	$n =$	$k =$
1	10	50	2×10^5
2	7	100	10^5
3	7	200	5×10^4
4	7	500	2×10^4
5	7	1000	10^4
6	7	2000	5000
7	10	3000	3000
8	7	5000	2000
9	7	10^4	1000
10	7	2×10^4	500
11	7	5×10^4	200
12	7	10^5	100
13	10	2×10^5	50

C. 爆燃

题目描述

小 α 给了你一个长度为 n 的序列 a_1, a_2, \dots, a_n 。小 α 有 q 个操作，每个操作为：

1. 给定 c ，令 $a_i \leftarrow \min(a_i, c)$ ，对每个 $1 \leq i \leq n$ 。
2. 给定 l, r, c ，令 $a_i \leftarrow \max(a_i, c)$ ，对每个 $l \leq i \leq r$ 。

一共有 $q!$ 种不同的操作顺序，对每种顺序，将它们依次操作到初始序列 a_1, a_2, \dots, a_n 后都能得到一个最终序列。问有多少种可能的不同最终序列，对 998244353 取模。

输入格式

从文件 `alpha.in` 中读入数据。

第一行三个正整数 n, m, k ，表示序列的长度，第一种操作的个数和第二种操作的个数。

第二行 n 个整数，表示初始序列 a_1, a_2, \dots, a_n 。

第三行 m 个整数，表示所有第一种操作的参数 c 。

接下来 k 行，每行三个整数 l, r, c ，表示一个操作。

输出格式

输出到文件 `alpha.out` 中。

一行一个正整数，表示最终序列个数对 998244353 取模的结果。

样例 1

样例 1 输入

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

样例 1 输出

```
6
```

附加样例 1 – 10

附加样例 1 – 10 输入

见选手目录下的 `alpha/ex_alpha1-10.in`。

附加样例 1 – 10 输出

见选手目录下的 `alpha/ex_alpha1-10.out`。

数据范围与说明

对于所有数据，保证 $1 \leq n, m, k \leq 150$ ， $1 \leq a_i \leq n$ ， $1 \leq l \leq r \leq n$ ， $1 \leq c \leq n$ 。

子任务编号	子任务分值	$n \leq$	$m \leq$	$k \leq$	特殊性质
1	10	20	3	7	
2	20		1		
3	20		5		
4	10				保证 $l = r$
5	20	30	30	30	
6	20				

D. 苦若吞沙

题目描述

小 ω 种了一排一共 n 株草，第 i 株草有一个原始高度 a_i ，还有一个目标高度 b_i 。

我们认为高度 > 0 的草是存活的，而高度 $= 0$ 的草已经死亡。

小 ω 想要修剪这些草，有两种修剪操作：

1. 选定一个区间 $[l, r]$ ，将 $[l, r]$ 内所有仍然存活的草的高度都增加 1；
 2. 选定一个区间 $[l, r]$ ，将 $[l, r]$ 内所有仍然存活的草的高度都削减 1；
- 如果一株草在经过这个操作后高度变为了 0，那么就被视为已经死亡，这株草不再受后续操作影响。

你需要求出让所有草的高度从 a_i 变成 b_i 所需要的修剪操作的最小个数。

输入格式

从文件 `omega.in` 中读入数据。

一个测试点中包含多组测试数据。

输入的第一行包含一个整数 T ，表示数据组数。

对于每组数据，第一行一个整数 n 。

接下来一行，包含 n 个整数，描述 a_1, a_2, \dots, a_n 。

接下来一行，包含 n 个整数，描述 b_1, b_2, \dots, b_n 。

输出格式

输出到文件 `omega.out` 中。

对于每组数据，输出一行一个整数，表示答案。若合法方案不存在，则输出 -1 。

样例 1

样例 1 输入

```
3
5
1 1 1 1 1
2 0 2 0 2
6
1 1 4 5 1 4
1 9 1 9 8 10
8
2 0 0 4 1 1 1 5
2 0 0 5 0 1 1 7
```

样例 1 输出

```
3
18
4
```

附加样例 1 – 10

附加样例 1 – 10 输入

见选手目录下的 `omega/ex_omega1-10.in`。

附加样例 1 – 10 输出

见选手目录下的 `omega/ex_omega1-10.out`。

数据范围与说明

对于所有数据，保证 $1 \leq n \leq 10^6$ ， $1 \leq \sum n \leq 3 \times 10^6$ ， $0 \leq a_i, b_i \leq 10^9$ 。
以下展示每个子任务各自的数据范围以及特殊性质：

子任务编号	子任务分值	$T \leq$	$n \leq$	$a_i, b_i \leq$	特殊性质
1	10		3		
2	10		4		
3	10		5		
4	10	10	100	100	
5	10	10	500	500	
6	10	10	3000	500	
7	10	10	10^5	500	
8	10				保证 $b_i > 0$
9	10				保证 a_i, b_i 单调不减
10	10				