

模拟赛

题目信息

序号	A	B	C	D
题目中文名	圣诞树	序列	绘画	聚会
题目英文名	tree	seq	paint	party
输入文件	tree.in	seq.in	paint.in	party.in
输出文件	tree.out	seq.out	paint.out	party.out
时间限制	2s	1s	2s	3s
空间限制	512MB	512MB	512MB	64MB
题目类型	传统题	传统题	传统题	传统题
子任务数量	5	4	3	6

注意事项：

- 编译选项： `-lm -O2 -std=c++14 -w1,--stack=998244353`
- 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是 `0`
- 若无特殊说明，结果比较方式为忽略行末空格，文末回车后的全文比较
- 所有题目采取子任务捆绑测试，选手只有通过一个子任务内的所有测试点才能获得该子任务的分数
- 题目不一定按难度顺序排列**，请选手自行判断
- 选手应将各题的源程序放在选手文件夹内，不要建立子文件夹
- 评测使用 Windows 系统，系统为64位
- 评测机配置：Intel(R) Core(TM) i7-8565U CPU @ 1.80GHz 1.99 GHz，内存 8G

A. 圣诞树 (tree)

题目描述

提示：本题输入量较大，请采用较快的读入方式

圣诞节到了，小P很孤独。

他只有一棵光秃秃的圣诞树，所以他决定玩玩它。不幸的是，这棵树被玩坏了，所以小P想将它复原。

小P的圣诞树是一棵 n 个节点的树，每个点是一个小球，第 i 个球上面有 d_i 个孔，**孔之间是有区别的**。小P需要用 $n - 1$ 条绳子将这些点连成一棵树。每个绳子应该连接两个不同的点的两个孔，且每个孔里最多有 1 根绳子。小P想知道，他能够复原出多少种不同的树。由于这个数太大了，他只想知道答案模 m 的结果。**注意 m 不一定为质数。**

- 定义两棵树相同当且仅当对于每一条边，它插入的两个孔在两棵树中相同，详见样例解释。

输入格式

第一行包含两个正整数 n, m 表示点数与模数。

第二行包含 n 个正整数，第 i 个表示 d_i 。

输出格式

一行一个整数，表示答案模 m 的值。

样例

样例 #1

输入

```
1 3 998244353
2 1 1 3
```

输出

```
1 6
```

样例解释

分三种情况讨论：

- 1 - 2 - 3，此时有 0 种方案，因为 2 号点要连两根绳子，但它只有一个孔。
- 2 - 1 - 3，此时有 0 种方案，因为 1 号点要连两根绳子，但它只有一个孔。
- 1 - 3 - 2，此时有 6 种方案，1 - 3 在 3 号点处有 3 种选择方式，2 - 3 在 3 号点处有剩余的 2 种选择方式，共有 $3 \times 2 = 6$ 种。

所以答案是 $0 + 0 + 6 = 6$

样例 #2

输入

1	6 123456789
2	7 3 5 10 6 4

输出

1	43617276
---	----------

样例 #3

输入

1	9 998244353
2	425656 453453 4320 1231 9582 54336 31435436 14342 423543

输出

1	667877982
---	-----------

数据范围与约定

对于所有数据, 满足 $2 \leq n \leq 10^6, 10^8 < m \leq 10^9, 1 \leq d_i < m$

子任务1 (10分) : 满足 $n \leq 5, d_i \leq 5$

子任务2 (20分) : 满足 $n \leq 5$

子任务3 (10分) : 满足 $d_i \leq 10$

子任务4 (30分) : 满足 $n \leq 2 \times 10^5, m = 998244353$

子任务5 (30分) : 无附加限制

B. 序列 (seq)

题目描述

小P找到了小T玩，他们在玩数列。

小T给了小P一个正整数序列，小P将它划分成了若干个连续子段，使得每段的元素都严格递增，并且小P很懒，所以他找到了划分段数最少的划分方案。

但是现在他一不小心把序列打乱了，小T很生气。为了安慰小T，他决定找回原来的序列。但他只记得每个元素出现了多少次，以及最后算出的最少划分段数。

现在他想知道，有多少种序列符合他的记忆呢？请求出这个值对 $10^9 + 7$ 取模的结果。由于小P记忆力不太好，所以他可能记忆有误，此时应当输出 0。

形式化地说，给定序列 $\{a_i\}$ ，表示在原序列中 i 出现了 a_i 次，再给定 k ，表示将原序列划分为最少的连续严格递增子段的个数，求可能的原序列个数，对 $10^9 + 7$ 取模。

输入格式

第一行两个整数 n, k 分别表示原序列值域与最少划分段数。

第二行 n 个整数，第 i 个整数 a_i 表示原序列中 i 的出现次数。

输出格式

一行一个整数，表示方案数对 $10^9 + 7$ 取模的结果。

样例

样例 #1

输入

```
1 | 3 2
2 | 1 1 1
```

输出

```
1 | 4
```

样例解释

有 4 种情况：

1. $\{1, 3, 2\}$ ，划分为 $\{1, 3\}, \{2\}$
2. $\{2, 1, 3\}$ ，划分为 $\{2\}, \{1, 3\}$
3. $\{2, 3, 1\}$ ，划分为 $\{2, 3\}, \{1\}$
4. $\{3, 1, 2\}$ ，划分为 $\{3\}, \{1, 2\}$

样例 #2

输入

1	5 20
2	3 2 11 5 7

输出

1	474640725
---	-----------

样例 #3

输入

1	2 30
2	28 14

输出

1	768379691
---	-----------

数据范围与约定

对于所有数据，满足 $1 \leq n \leq 50, 1 \leq k \leq 2500, 1 \leq a_i \leq 50$

子任务1（5分）：满足 $n \leq 2$

子任务2（20分）：满足 $n \leq 5, \sum a_i \leq 10$

子任务3（25分）：满足 $\sum a_i \leq 20$

子任务4（50分）：无附加限制

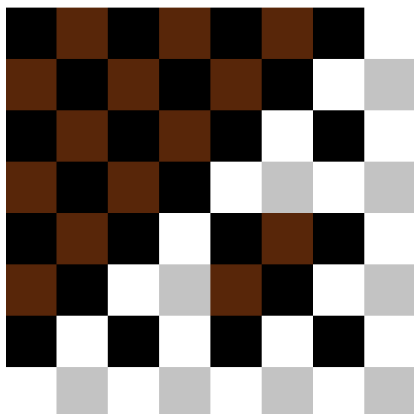
C. 绘画 (paint)

题目描述

可惜小P没能算出答案，小T很生气，所以离开了他。他现在和小Q学画画。

小Q首先教他如何使用画板。一个画板是一个 $2^k \times 2^k$ 的方格纸，从左到右，从下到上分别编号为 $0 \sim 2^k - 1$ ，有一些格子是白色，另一些是黑色。

我们定义一个 $2^k \times 2^k$ 的画板是“ k 阶”的。一个 0 阶的画板就是一个白格子，一个 k ($k > 0$) 阶的画板由四个 $2^{k-1} \times 2^{k-1}$ 小画板拼成，其中左上角的全是黑格子，右上角、左下角、右下角是 $k-1$ 阶画板。如图所示的是一个 3 阶画板（深色的格子表示黑色，浅色的表示白格子）。



现在小Q将两个 k 阶画板叠在一起，其中一个画板的左下角叠在另一个画板的 (X, Y) 位置上，她想让小P求两个画板重叠部分中在两个画板中都是白格子的位置的数量，**不取模**。

输入格式

一行三个整数， k, X, Y ，分别表示画板的阶数与重叠的位置。

输出格式

一行一个整数，表示答案，**不取模**

样例

样例 #1

输入

```
1 | 2 2 2
```

输出

```
1 | 3
```

样例解释

假设我们以下面画板的左下角为 $(0, 0)$ ，那么：

第一个画板上位置 $(0, 0), (1, 0), (1, 1), (2, 0), (3, 0), (3, 1), (2, 2), (3, 2), (3, 3)$ 都是白色，

第二个画板上位置 $(2, 2), (3, 2), (3, 3), (4, 2), (5, 2), (5, 3), (4, 4), (5, 4), (5, 5)$ 都是白色，

二者的重合部分为 $(2, 2), (3, 2), (3, 3)$ ，共有 3 个位置。

样例 #2

输入

1 | 4 1 3

输出

1 | 21

样例 #3

输入

1 | 80 7285610927563948192653 98210647293847192059

输出

1 | 10352717033437218767116074082

不取模!

数据范围与约定

对于所有数据，满足 $1 \leq k \leq 5000, 0 \leq X, Y \leq 2^k$

子任务1 (20分)：满足 $k \leq 10$

子任务2 (30分)：满足 $k \leq 100$

子任务3 (50分)：无附加限制

D. 聚会 (party)

题目描述

提示：注意本题不寻常的空间限制

小P因为没有艺术细胞而被小Q嫌弃，不得不另谋生路。他为了能够交到朋友，参加了一个社团。

这个社团有 n 个成员，初始时大家两两互不认识。现在第一天大家在某段时间内来社团总部参加了活动。我们设第 i 个人在 s_i 时刻到来，在 t_i 时刻离开。如果两个人在同一时刻都在社团里，那这两个人会在社交软件上互相添加为好友，这包括在一个人到来时另一个人恰好离开的情况（即若 $s_i = t_j$ ，那 i 和 j 同样会加好友）。

第二天，每个人在社交软件上发表了一篇自我介绍。这个社交软件的规则是每个人可以看到他好友发表的一切东西（包括转发的内容），也可以转发他的好友发表的东西，但一次只能转发一条消息。

由于每个人都和小P一样交友心切，他们都想看到所有人发表的自我介绍。当然，这可能需要一部分进行转发。小P作为社长，希望在满足所有人愿望的情况下，所有人转发次数的总数尽量少，现在他要求这个总数。注意要求的是转发次数，即一开始发表的那次不算在内。如果不能满足所有人的要求，输出 -1 。

输入格式

第一行一个整数 n ，表示人数。

下面 n 行，第 i 行两个整数 s_i, t_i ，表示第 i 个人在 s_i 时刻到达，在 t_i 时刻离开。

输出格式

一行一个整数，表示最少的转发次数。

样例

样例 #1

输入

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

输出

```
1 | 2
```

解释

第一个人与第二个人是好友，第二个人与第三个人是好友。

第二个人的消息另外两个人都能看到，不需要转发。

第一个人的消息第三个人看不到，需要第二个人转发 1 次。

第三个人的消息第一个人看不到，需要第二个人转发 1 次。

所以总共需要转发 2 次。

样例 #2

输入

1	3
2	1 5
3	3 8
4	10 12

输出

1	-1
---	----

样例 #3

输入

1	10
2	7 9
3	6 9
4	2 2
5	6 7
6	4 6
7	6 8
8	3 6
9	1 4
10	4 7
11	6 9

输出

1	18
---	----

数据范围与约定

对于所有数据，满足 $1 \leq n \leq 2 \times 10^5, 1 \leq s_i \leq t_i \leq 10^9$

子任务1 (5分)：满足 $n \leq 20$

子任务2 (15分)：满足 $n \leq 200$

子任务3 (10分)：满足 $n \leq 1000$

子任务4 (30分)：满足 $n \leq 5000$

子任务5 (10分)：满足对于所有 $1 \leq i < n$ ，都有 $s_i < s_{i+1}, t_i < t_{i+1}$

子任务6 (30分)：无附加限制