

云斗 CSP-J 模拟赛

时间：2025 年 10 月

题目名称	不平凡数	文物修复	抽奖	芳香串
题目类型	传统型	传统型	传统型	传统型
可执行文件名	num	fix	draw	string
时间限制	2.0 秒	1.0 秒	1.0 秒	2.0 秒
内存限制	512 MiB	256 MiB	256 MiB	256 MiB
测试点数目	10	20	10	25
测试点等分	是	是	是	是

提交源程序文件名

对于 C++ 语言	num.cpp	fix.cpp	draw.cpp	string.cpp
-----------	---------	---------	----------	------------

编译选项

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

注意事项：

- C++ 中函数 main() 的返回值类型必须是 int，值必须为 0。
- 若无特殊说明，输入文件中同一行内的多个整数、浮点数、字符串等均使用一个空格进行分隔。
- 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。
- 程序可使用的栈空间大小与该题内存空间限制一致。
- 在终端下可使用命令 `ulimit -s unlimited` 将栈空间限制放大，但你使用的栈空间大小不应超过题目限制。
- 对于因未遵守以上规则对成绩造成的影响，相关申诉不予受理。
- 线上提交渠道：[比赛链接](#)。注意，线上提交也应采用文件输入输出。



不平凡数 (num)

【题目描述】

如果一个正整数 N 能够被表示成 $x^2 - y^2$ 的形式 (x, y 都是 **非负整数**)，则称 N 是不平凡的。

给出 T 个正整数，请判断这 T 个正整数是不是不平凡的。

【输入格式】

第一行一个正整数 T ；

接下来 T 行，每行一个正整数 N ，表示一次查询。

【输出格式】

输出共 T 行，每一行代表一次询问的结果。对于每次询问，如果 N 是不平凡的，输出 YES，否则输出 NO。

【样例输入 1】

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

【样例输出 1】

```
1 NO
2 YES
3 YES
```

【样例解释】

$3 = 2^2 - 1^2$, $4 = 2^2 - 0^2$ 。

【数据范围和约定】

对于 30% 的数据， $1 \leq N \leq 5000$ ；
对于 60% 的数据， $1 \leq N \leq 10^6$ ；
对于另外 20% 的数据， N 是奇数；
对于 100% 的数据， $1 \leq T \leq 5 \times 10^6$ ， $1 \leq N \leq 10^9$ 。

文物修复 (fix)

【题目描述】

在市中心的古籍修复室里，静静存放着 N 件待修复的珍贵文物，每件文物都因受损程度不同，标注着对应的修复难度评分 D_i 。要成功修复一件文物，你的修复工具熟练度必须至少达到该文物的难度评分——可刚接手任务时，你的工具熟练度还处于初始的 0 分状态，想要让所有文物重焕光彩，得好好规划每天的修复节奏。

在所有文物修复完成前，你的每天都会按固定流程推进：

- 每天清晨，你必须从两种方式中选择一种来调整工具熟练度：
 - 选择“工具校准”：花费时间仔细模拟使用修复工具，让工具熟练度直接提升 x 分；
 - 选择“工具保养”：简单清洁工具以延长使用寿命，但工具熟练度会随之降低 x 分（熟练度甚至可能变成负数）。
- 每天下午，在完成清晨的工具调整后，你可以挑选一件文物尝试修复，但有个关键前提：此时你的工具熟练度必须大于或等于这件文物的难度评分 D_i ，否则无法开展修复工作，当然也可以选择下午偷懒不修复。

你并不着急赶工期，不在乎总共需要多少天完成修复，但特别喜欢“工具保养”时的轻松状态，所以核心目标是：尽可能减少清晨选择“工具校准”的次数，用最少的校准次数，完成所有文物的修复任务。现在需要你规划好工具校准、保养和文物修复的顺序，找出修复所有文物所需的最少“工具校准”次数。

【输入格式】

输入共两行。

第一行包含两个整数 N, x ，分别代表待修复的文物总数、每次“工具校准”提升的熟练度分值或每次“工具保养”降低的熟练度分值。

第二行包含 N 个整数，描述每件文物的修复难度水平，其中第 i 个整数代表第 i 件文物的难度评分 D_i 。

【输出格式】

输出一个整数，代表修复所有 N 件文物所需的最少“工具校准”早晨次数。

【样例输入 1】

```
1 5 2
2 5 3 7 1 3
```

【样例输出 1】

```
1 4
```

【样例输入 2】

```
1 4 3  
2 5 9 10 2
```

【样例输出 2】

```
1 4
```

【样例输入 3】

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

【样例输出 3】

```
1 9
```

【数据范围和约定】

对于 10% 的数据，满足 $1 \leq n \leq 10$ 。

对于 60% 的数据，满足 $1 \leq n \leq 10^3$ 。

对于 100% 的数据，满足 $1 \leq n, x, D_i \leq 10^5$ 。

抽奖 (draw)

【题目背景】

在你的帮助下小 U 成功获得了这次知识大赛的总冠军。
但坑人的主办方决定用抽奖的方式发放奖品——星球。

【题目描述】

抽奖规则是这样的：

一共有 n 个星球，第 i 个星球有一个价值 a_i 。

将由选手选出一个抽奖区间，记为 $[l, r]$ 。

接下来主办方将在 $[l, r]$ 中随机选取两个数（两次选取互不干扰）记为 x, y 。

而选手将获得编号在 $[\min(x, y), \max(x, y)]$ 之中的星球。

小 U 希望聪明的你告诉他，如果他选取的区间为 $[l, r]$ 那么他获得的星球总价值的所有情况的和是多少。

即

$$\sum_{i=\min(x,y)}^{\max(x,y)} a_i$$

的和是多少（对于所有 x, y 组合）。

小 U 一共会问你 m 次。

注意：

由于答案过大请对 $10^9 + 7$ 取余。

为了减少输出时间请输出 m 次询问的和对 $10^9 + 7$ 的余数。

为了减少读入时间本题由数据生成器生成数据，具体格式见 [sample.cpp](#)。

【输入格式】

第一行三个整数， n, m, sd 。

【输出格式】

共一行一个整数表示最终答案。

【样例输入 1】

```
1 10 2 10
```

【样例输出 1】

1 330468653

【样例输入 2】

1 50000 5000 1236998

【样例输出 2】

1 137944585

【样例输入 3】

1 100000 100000 124701

【样例输出 3】

1 608177768

【样例解释】

具体数据如下：

1 646587008 172788309 748409565 431728518 891360240 964941590
694380241 566875052 705025035 813088567
2 8 8
3 2 3

两次答案如下：

566875052

763593608

【数据范围】

对于 20% 的数据， $1 \leq n, m \leq 500$ 。

对于 40% 的数据， $1 \leq n, m \leq 5 \times 10^3$ 。

对于 60% 的数据， $1 \leq n, m \leq 10^5$ 。

对于 80% 的数据， $1 \leq n, m \leq 10^6$ 。

对于 100% 的数据，

$$1 \leq l \leq r \leq n, \quad m \leq 5 \times 10^6, \quad 1 \leq a_i \leq 10^9$$

芳香串 (string)

【题目描述】

小 M 家里有一个字符串 $s = "114514"$ 。

定义函数：

$$\text{cmp}(x, y) = \begin{cases} 1, & x = y \\ 0, & x \neq y \end{cases}$$

给出一个字符串 t , 记 t_i 表示字符串的第 i 个字符。对于一个由 t 中 6 个字符 $t_{n_0}, t_{n_1}, \dots, t_{n_5}$ 依次拼接组成的字符串 t' ($n_0 < n_1 < n_2 < n_3 < n_4 < n_5$), 当满足

$$\forall 0 \leq i < j \leq 5, \quad \text{cmp}(t_{n_i}, t_{n_j}) = \text{cmp}(s_i, s_j)$$

时, 称 t' 为一个芳香串。

小 M 想求出：有多少种选择下标的方式，使得选出的字符串 t' 为芳香串？

答案对 998244353 取模。

【输入格式】

一行一个字符串 t 。

【输出格式】

一行一个整数，表示答案。

【样例输入 1】

```
1 11451419911225
```

【样例输出 1】

```
1 8
```

【样例解释】

我们能找到 114514, 115415, 115915 共计 3 个芳香串。但是一共有 8 个下标选择方案满足选出来的串是芳香串。

【数据范围】

设字符串 t 的长度为 n ($n \geq 6$)， t 中出现的不同字符数量为 m 。

测试点编号	$n \leq$	$m \leq$
1-3	12	10
4-6	100	10
7-9	2000	10
10-14	10^5	62
15-16	5×10^5	3
17-19	5×10^5	10
20-21	5×10^5	62
22-25	10^6	62

保证 t 只由大写字母、小写字母、数字组成。附加样例请见附件。