

划分 (partition)

【题目描述】

2048 年，第三十届 CSP 认证的考场上，作为选手的小明打开了第一题。这个题的样例有 n 组数据，数据从 $1 \sim n$ 编号， i 号数据的规模为 a_i 。

小明对该题设计出了一个暴力程序，对于一组规模为 u 的数据，该程序的运行时间为 u^2 。然而这个程序运行完一组规模为 u 的数据之后，它将在任何一组规模小于 u 的数据上运行错误。样例中的 a_i 不一定递增，但小明又想在不修改程序的情况下正确运行样例，于是小明决定使用一种非常原始的解决方案：将所有数据划分成若干个数据段，段内数据编号连续，接着将同一段内的数据合并成新数据，其规模等于段内原数据的规模之和，小明将让新数据的规模能够递增。

也就是说，小明需要找到一些分界点 $1 \leq k_1 < k_2 < \dots < k_p < n$ ，使得

$$\sum_{i=1}^{k_1} a_i \leq \sum_{i=k_1+1}^{k_2} a_i \leq \dots \leq \sum_{i=k_p+1}^n a_i$$

注意 p 可以为 0 且此时 $k_0 = 0$ ，也就是小明可以将所有数据合并在一起运行。

小明希望他的程序在正确运行样例情况下，运行时间也能尽量小，也就是最小化

$$\left(\sum_{i=1}^{k_1} a_i\right)^2 + \left(\sum_{i=k_1+1}^{k_2} a_i\right)^2 + \dots + \left(\sum_{i=k_p+1}^n a_i\right)^2$$

小明觉得这个问题非常有趣，并向你请教：给定 n 和 a_i ，请你求出最优划分方案下，小明的程序的最小运行时间。

【输入格式】

从文件 `partition.in` 中读入数据。

由于本题的数据范围较大，部分测试点的 a_i 将在程序内生成。

第一行两个整数 $n, type$ 。 n 的意义见题目描述， $type$ 表示输入方式。

1. 若 $type = 0$ ，则该测试点的 a_i 直接给出。输入文件接下来：第二行 n 个以空格分隔的整数 a_i ，表示每组数据的规模。
2. 若 $type = 1$ ，则该测试点的 a_i 将特殊生成，生成方式见后文。输入文件接下来：第二行六个以空格分隔的整数 x, y, z, b_1, b_2, m 。接下来 m 行中，第 i ($1 \leq i \leq m$) 行包含三个以空格分隔的正整数 p_i, l_i, r_i 。

对于 $type = 1$ 的 23 ~ 25 号测试点， a_i 的生成方式如下：

给定整数 x, y, z, b_1, b_2, m ，以及 m 个三元组 (p_i, l_i, r_i) 。

保证 $n \geq 2$ 。若 $n > 2$ ，则 $\forall 3 \leq i \leq n$ ， $b_i = (x \times b_{i-1} + y \times b_{i-2} + z) \bmod 2^{30}$ 。

保证 $1 \leq p_i \leq n$ ， $p_m = n$ 。令 $p_0 = 0$ ，则 p_i 还满足 $\forall 0 \leq i < m$ 有 $p_i < p_{i+1}$ 。

对于所有 $1 \leq j \leq m$, 若下标值 i ($1 \leq i \leq n$) 满足 $p_{j-1} < i \leq p_j$, 则有

$$a_i = (b_i \bmod (r_j - l_j + 1)) + l_j$$

上述数据生成方式仅是为了减少输入量大小, 标准算法不依赖于该生成方式。

【输出格式】

输出到文件 *partition.out* 中。

输出一行一个整数, 表示答案。

【样例 1 输入】

```
5 0
5 1 7 9 9
```

【样例 1 输出】

```
247
```

【样例 1 解释】

最优的划分方案为 $\{5,1\},\{7\},\{9\},\{9\}$ 。由 $5+1 \leq 7 \leq 9 \leq 9$ 知该方案合法。

答案为 $(5+1)^2 + 7^2 + 9^2 + 9^2 = 247$ 。

虽然划分方案 $\{5\},\{1\},\{7\},\{9\},\{9\}$ 对应的运行时间比 247 小, 但它不是一组合法方案, 因为 $5 > 1$ 。

虽然划分方案 $\{5\},\{1,7\},\{9\},\{9\}$ 合法, 但该方案对应的运行时间为 251, 比 247 大。

【样例 2 输入】

```
10 0
5 6 7 7 4 6 2 13 19 9
```

【样例 2 输出】

```
1256
```

【样例 2 解释】

最优的划分方案为 $\{5\},\{6\},\{7\},\{7\},\{4,6,2\},\{13\},\{19,9\}$ 。

【样例 3 输入】

```

10000000 1
123 456 789 12345 6789 3
2000000 123456789 987654321
7000000 234567891 876543219
10000000 456789123 567891234

```

【样例 3 输出】

```

4972194419293431240859891640

```

【样例 4】

见选手目录下的 *partition/partition4.in* 与 *partition/partition4.ans*。

【样例 5】

见选手目录下的 *partition/partition5.in* 与 *partition/partition5.ans*。

【数据范围】

测试点编号	$n \leq$	$a_i \leq$	$type =$
1 ~ 3	10	10	0
4 ~ 6	50	10^3	
7 ~ 9	400	10^4	
10 ~ 16	5000	10^5	
17 ~ 22	5×10^5	10^6	
23 ~ 25	4×10^7	10^9	1

所有测试点满足： $type \in \{0, 1\}$ ， $2 \leq n \leq 4 \times 10^7$ ， $1 \leq a_i \leq 10^9$ ， $1 \leq m \leq 10^5$ ， $1 \leq l_i \leq r_i \leq 10^9$ ， $0 \leq x, y, z, b_1, b_2 < 2^{30}$ 。