

# 2026 第五届昆明青少年人工智能综合实践大赛

## Python 编程决赛（入门组）

考试时间：2026 年 4 月 25 日 8:30-12:00

题目名称	准备晚饭	整齐队伍	挖通湖泊	封印
题目类型	传统型	传统型	传统型	传统型
目录	dinner	group	dig	seal
可执行文件名	dinner	group	dig	seal
输入文件名	dinner.in	group.in	dig.in	seal.in
输出文件名	dinner.out	group.out	dig.out	seal.out
每个测试点时限	1.0 秒	1.0 秒	1.0 秒	1.0 秒
内存限制	128MB	128MB	256MB	256MB
子任务数目	10	10	10	20
测试点是否等分	是	是	是	是

提交源程序文件名

对于 Python 语言	dinner.py	group.py	dig.py	seal.py
--------------	-----------	----------	--------	---------

### 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. 提交的程序代码文件的放置位置请参考文件提交格式的具体要求。
3. 因违反以上三点而出现的错误或问题，申诉时一律不予受理。
4. 若无特殊说明，结果的比较方式为全文比较（过滤行末空格及文末回车）。
5. 全省统一评测时采用的机器配置为：Intel(R) Core(TM) i7-8550U CPU @ 1.80GHz 1.99 GHz 16G 内存。上述时限以此配置为准。
6. 只提供 Linux 格式附加样例文件。
7. 评测在当前最新公布的 NOI Linux 下进行，各语言的编译器版本以此为准。

# 准备晚饭 (dinner)

## 【题目描述】

小向正在准备晚饭。

小向的电饭煲把饭煮熟需要 $a$ 分钟。小向需要准备 $n$ 个菜，其中第 $i$ 个菜都要先清洗食材花费 $b_i$ 分钟，再进行烹饪花费 $c_i$ 分钟，小向同一时刻只能烹饪一个菜。

求小向准备好晚饭需要花费多少时间。

## 【输入格式】

从文件 `dinner.in` 中读入数据。

第一行四个正整数 $n, a$ ，分别表示准备的菜的数量，电饭煲的煮饭时间。

接下来 $n$ 行，每行两个数字 $b_i, c_i$ ，表示第 $i$ 个菜洗食材花费时间、烹饪花费分钟。

## 【输出格式】

输出到文件 `dinner.out` 中。

输出仅一个整数，表示小向准备好晚饭花费的时间。

## 【样例 1 输入】

```
1 1 5
2 3 6
```

## 【样例 1 输出】

```
1 9
```

## 【样例 1 解释】

电饭煲煮饭需要花费 5 分钟，小向只需要准备一个菜，花费时间为  $3+6=9$ 。小向只要做好菜，电饭煲就煮好饭了。

## 【样例 2 输入】

```
1 2 20
2 1 2
3 3 4
```

**【样例 2 输出】**

```
1 20
```

**【样例 2 解释】**

电饭煲煮饭需要花费 20 分钟，小向需要准备两个菜，花费时间为  $1+2+3+4=10$ 。小向做好菜还需要等电饭煲煮完饭。

**【数据范围】**

对于所有测试数据保证： $1 \leq n \leq 10^5, 1 \leq a, b_i, c_i \leq 10^6$ 。

测试点	$n \leq$	$a, b_i, c_i \leq$
1~2	10	$10^6$
3~5	$10^3$	$10^6$
6~10	$10^5$	$10^6$

# 整齐队伍 (group)

## 【题目描述】

三年 A 班准备出早操了,  $n$  位同学们熙熙攘攘的从教室出来, 从左到右排成一排。不过同学们的位置是随便选的, 惹得班主任向老师非常生气。

向老师命令他们重新从矮到高排成一排, 在所有同学都知道自己目标位置情况下, 所有同学需要最少总共移动多少步? 已知目前两两相邻的同学之间的距离为 1 步。

## 【输入格式】

从文件 `group.in` 中读入数据。

第一行包含 1 个数字  $n$ , 表示同学的数量。

接着  $n$  行, 每行一个正整数, 表示目前从左到右的每个同学的身高。

## 【输出格式】

输出到文件 `group.out` 中。

输出仅 1 个数字, 表示同学需要移动的总步数。

## 【样例 1 输入】

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

## 【样例 1 输出】

```
1 4
```

## 【样例 1 解释】

有 4 位同学, 从左到右身高分别是 3、1、2、4。从矮到高重排后身高分别是 1、2、3、4。

身高为 3 的同学需要移动 2 步到目标位置, 身高为 1 的同学需要移动 1 步

到目标位置，身高为 2 的同学需要移动 1 步到目标位置。总共需要 4 步。

**【样例 2】**

见选手目录下 `group / group2.in` 和 `group / group2.ans`。

**【样例 3】**

见选手目录下 `group / group3.in` 和 `group / group3.ans`。

**【数据范围】**

对于所有测试数据有： $1 \leq n \leq 10^5, 1 \leq a_i \leq 10^5$ 。

测试点	$n \leq$
1~2	10
3~5	$10^3$
6~10	$10^5$

# 挖通湖泊 (dig)

## 【题目描述】

X 国地理特征独特，用一个  $n$  行  $m$  列的二维网格表示。X 国内有两个湖泊，其中一个湖泊在春夏水位暴涨，在秋冬水位下降；另外一个则是在秋冬水位暴涨，在春夏水位下降。一个湖泊由上下左右相邻的字符 '0' 网格组成。除了湖泊的网格外，都是地面网格，地面网格都是用字符 '1' 表示。

X 国为了保障国内的农耕活动，决定把两个湖泊挖通，以平衡水位。不过耕地面积也很重要，X 国希望能够保留尽可能多的地面网格数量。

于是这个重要的任务就落到了你身上，求在挖通两个湖泊后，X 国能保留的最大地面网格数。

## 【输入格式】

从文件 **dig.in** 中读入数据。

第一行 2 个整数  $n$ 、 $m$ ，表示网格的行数与列数。

接着  $n$  行，每行  $m$  个字符，其中字符 0 表示湖泊网格、字符 1 表示地面网格。

## 【输出格式】

输出到文件 **dig.out** 中。

输出仅 1 个整数，表示挖通两个湖泊后，X 国能保留的最大地面网格数。

## 【样例 1 输入】

```
1 4 4
2 0111
3 1111
4 1111
5 1110
```

## 【样例 1 输出】

```
1 9
```

**【样例 1 解释】**

二维网格情况如下图：

湖	地	地	地
地	地	地	地
地	地	地	地
地	地	地	湖

挖通左上角的湖和右下角的湖至少需要 5 格，方案有很多，下面给出其中一种：

湖	湖	湖	湖
地	地	地	湖
地	地	地	湖
地	地	地	湖

剩下地面的网格数是 9 格。

**【样例 2 输入】**

```
1 5 5
2 00000
3 01110
4 01010
5 01110
6 00000
```

**【样例 2 输出】**

1 7

**【样例解释】**

二维网格如下图。

湖1	湖1	湖1	湖1	湖1
湖1	地	地	地	湖1
湖1	地	湖2	地	湖1
湖1	地	地	地	湖1
湖1	湖1	湖1	湖1	湖1

只要挖掉湖 2 那个网格上、下、左、右相邻的其中任意一个网格就可以把两个湖连通，此时地面网格数为 7。

**【数据范围】**

对于所有测试数据保证： $1 \leq n, m \leq 2000$ 。保证只有两个湖泊，保证两个湖泊起始不连通。

测试点	$n, m \leq$
1~2	20
3~4	100
5~6	200
7~8	1000
9~10	2000

# 封印 (seal)

## 【题目描述】

在远古的神话时代，众神曾创造了一件拥有无限力量的神器——“万物之核”。为了防止凡人滥用这股力量，众神将它的力量分解为 $n$ 个圣数，刻在一块名为“命运之碑”的石板上，它们分别是 $a_1, a_2, \dots, a_n$ 。

传说中，若要重新唤醒这件神器，必须将这 $n$ 个圣数相乘，得到“本原之数”：

$$N = \prod_{1 \leq i \leq n} a_i = a_1 \times a_2 \times \dots \times a_n$$

而神器的封印能否解开，取决于一个更深的奥秘——本原之数的所有正因子之和。只有当这个和等于某个天启之数时，封印才会显现裂痕。

这个数字可能很大，人类很难计算，众神决定给人类一个破解的机会：结果必须对 $10^9 + 7$ 取模。

如今，你在古神殿的废墟中发现了这块命运之碑。作为一位精通数论的探险者，你必须迅速计算：

$$\sigma(N) = \left( \sum_{d|N} d \right) \bmod (10^9 + 7)$$

即 $N$ 的所有正因子之和，再对 $10^9 + 7$ 取模。

## 【输入格式】

从文件 `seal.in` 中读入数据。

输入的第一行为数字 $n$ 。

输入的第二行包含 $n$ 个数字，分别为 $a_1, a_2, \dots, a_n$ 。

## 【输出格式】

输出到文件 `seal.out` 中。

输出仅一个数字，即 $N$ 的所有正因子之和，再对 $10^9 + 7$ 取模的结果。

## 【样例 1 输入】

```
1 3
2 2 5 10
```

**【样例 1 输出】**

```
1 217
```

**【样例 1 解释】**

$N = 2 \times 5 \times 10 = 100$ 。100 的因子有：1、2、4、5、10、20、25、50、100。  
总和为  $1 + 2 + 4 + 5 + 10 + 20 + 25 + 50 + 100 = 217$ 。

**【数据范围】**

对于所有测试数据保证： $1 \leq n, a_i \leq 2 \times 10^5$

测试点	$n \leq$	特殊性质
1~3	10	$a_i \leq 5$
4~5	$10^2$	无
6	$10^3$	$a_i \leq 5$
7~8	$10^3$	无
9	$10^4$	$a_i$ 为两两不同的质数
10	$10^4$	无
11~12	$10^5$	无
13~14	$10^5$	$a_i \leq 2$
15~16	$2 \times 10^5$	$a_i = 2^k, k$ 为正整数
17~20	$2 \times 10^5$	无