

苏州市人工智能学会

青少年人工智能核心算法素养考核 (SACCC)

3 级

时间：2026 年 2 月 7 日 08:30 ~ 2027 年 2 月 7 日 11:10

题目名称	算式	可乐	下棋	真人秀	飞船	翻滚的长方体
题目类型	传统型	传统型	传统型	传统型	传统型	传统型
每个测试点时限	1.0 秒					
内存限制	256 MiB					
子任务数目	5	5	5	5	5	5

编译选项

对于 C++ 语言	<code>-std=c++14 -O2 -static</code>
-----------	-------------------------------------

算式 (express)

【题目描述】

人工智能也离不开最简单的加减乘除运算。请你设计一个程序，实现以下功能：从键盘输入任意 n 个整数的加法、减法表达式，计算并输出结果。

【输入格式】

从文件 `express.in` 中读入数据。

输入共一行，可能包含加号 (+)、减号 (-)、数字的表达式，表达式长度不超过 50，且出现的整数位数不超过 5 位，表达式中间没有空格和括号。

【输出格式】

输出到文件 `express.out` 中。

输出一行一个整数，表示表达式的计算结果。

【样例 1 输入】

```
1 1-2+3
```

【样例 1 输出】

```
1 2
```

【数据范围】

- 对于 20% 的数据，保证 $n \leq 2$ 。
- 对于 60% 的数据，保证 $n \leq 3$ 。
- 对于全部测试数据，保证： $1 \leq n \leq 10$ ，表达式的长度不超过 50，且其中出现的整数位数不超过 5 位。

可乐 (coco)

【题目描述】

小智去超市买可乐，正赶上超市在搞活动。每 k 个可乐瓶可以换一瓶新的可乐。小智一共买了 n 瓶可乐，若超市的活动一直持续下去，问小智最多能喝到多少瓶可乐。

【输入格式】

从文件 `coco.in` 中读入数据。

输入共一行，包括两个整数 n 、 k 。

【输出格式】

输出到文件 `coco.out` 中。

输出一行一个整数，表示小智最多能喝到的可乐瓶数。

【样例 1 输入】

```
1 4 3
```

【样例 1 输出】

```
1 5
```

【样例 2 输入】

```
1 10 3
```

【样例 2 输出】

```
1 14
```

【数据范围】

- 对于 20% 的数据，保证： $n \leq 10$ ， $k \leq 10$ 。
- 对于 60% 的数据，保证： $n \leq 100$ ， $k \leq 100$ 。
- 对于全部测试数据，保证： $1 \leq n \leq 10^8$ ， $1 \leq k \leq 1000$ 。

下棋 (win)

【题目描述】

小能和机器人小智玩一个博弈游戏。在一个 $n \times n$ 的棋盘上，一个石头被放在棋盘的左上角。他们轮流移动石头。每一回合，选手只能把石头向上，下，左，右四个方向移动一格，并且要求被之前石头走过的格子之后不能再放第二次。谁先不能移动石头了就算输。

假如小能先移动石头，且两个选手都以最优策略走步，问最后谁能赢？

【输入格式】

从文件 `win.in` 中读入数据。

本题的输入包含多组测试数据。

每组测试数据共一行，输入一个整数 n ，表示棋盘的规模。

当输入的 n 为 0 时，表示输入结束。

【输出格式】

输出到文件 `win.out` 中。

对于每组数据，如果小能最后能赢，则输出 `neng`，否则输出 `zhi`，每一组答案独占一行。

【样例 1 输入】

```
1 2
2 0
```

【样例 1 输出】

```
1 neng
```

【数据范围】

- 对于 20% 的数据，保证： $0 \leq n \leq 10$ 。
- 对于 60% 的数据，保证： $0 \leq n \leq 3000$ 。
- 对于全部测试数据，保证： $0 \leq n \leq 10000$ 。

真人秀 (tvshow)

【题目描述】

小智报名参加了一个真人秀节目，这个节目设定了起点和终点，起点和终点之间有 q 个关卡，要求每个选手把 q 个关卡的密码带到终点。组委会给每位选手一本全是数字串的字典，并且每到达一个关卡，都会告知本关卡密码是以什么数字结尾的。小智的任务是从密码字典中找到匹配的数字串，并且符合条件的最小数字串即为关卡密码，如果密码字典中找不到，则输出 -1 表示不存在。

【输入格式】

从文件 `tvshow.in` 中读入数据。

第一行两个正整数 n 、 q ，分别代表密码字典中密码串的数量和关卡的数量。

接下来的 n 行，每行包含一个正整数，代表密码字典中的密码。

接下来的 q 行，每行包含两个正整数，第一个正整数表示已知关卡末尾数字串的长度，第二个正整数表示关卡末尾数字串。

【输出格式】

输出到文件 `tvshow.out` 中。

输出 q 行，每行一个整数，如果存在第 i 个关卡的密码，则在第 i 行输出第 i 个关卡的完整密码，否则输出 -1 。

【样例 1 输入】

```
1 5 5
2 2123
3 1123
4 23
5 24
6 24
7 2 23
8 3 123
9 3 124
10 2 12
11 2 12
```

【样例 1 输出】

```
1 23
2 1123
3 -1
4 -1
5 -1
```

【样例 2 输入】

```
1 1 10
2 9392912
3 3 912
4 2 10
5 7 2731197
6 2 12
7 7 7163045
8 4 8785
9 7 4577297
10 5 75400
11 1 2
12 6 493284
```

【样例 2 输出】

```
1 9392912
2 -1
3 -1
4 9392912
5 -1
6 -1
7 -1
8 -1
9 9392912
10 -1
```

【数据范围】

- 对于 20% 的数据, 保证: $n \leq 10, q \leq 20$ 。
- 对于 60% 的数据, 保证: $n \leq 500, q \leq 500$ 。
- 对于全部测试数据, 保证: $1 \leq n \leq 1000, 1 \leq q \leq 1000$ 。

飞船 (ship)

【题目描述】

小智的飞船有左右两个发动机，左右两个发动机同时工作才能保证飞船稳定前进，每个燃料瓶同时只能供一台发动机使用。为了保证尽量长的稳定飞行时间，他买了 n 个发动机专用燃料瓶，这些燃料瓶的使用寿命不完全相同，分别用正整数 a_i 表示燃料瓶可以持续使用多少个小时。

现在已知燃料瓶的数量和燃料瓶能够使用的时间，请你找一种方案使得飞船稳定前进的时间尽可能的长。

【输入格式】

从文件 `ship.in` 中读入数据。

本题的输入包含多组测试数据。

每组测试数据共两行，第一行一个整数 n ，表示燃料瓶的数目；第二行 n 个正整数 a_i ，表示燃料瓶能使用的时间。

【输出格式】

输出到文件 `ship.out` 中。

每组数据输出一行，表示飞船能稳定前进的最大时间，保留到小数点后 1 位。

【样例 1 输入】

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

【样例 1 输出】

```
1 3.0
2 5.5
```

【样例 1 解释】

第一组数据他只有两个燃料瓶，一个能用 5 小时一个能用 3 小时，那么他的飞船只能稳定飞行 3 个小时，有一个燃料瓶剩下的燃料无法使用，但是如果他有更多的燃料瓶，就可以更加充分地利用它们。

第二组数据 he 有三个燃料瓶，分别能用 3、3、5 小时， he 可以先使用两节能用 3 个小时的燃料瓶，使用半个小时后再把其中一个换成能使用 5 个小时的燃料瓶，两个半小时后再把剩下的一个燃料瓶换成刚才换下的燃料瓶（那个燃料瓶还能用 2.5 个小时），这样总共就可以使用 5.5 个小时，没有一点浪费。

【数据范围】

- 对于 20% 的数据，保证： $n \leq 5$ ， $0 \leq a_i \leq 10$ 。
- 对于 60% 的数据，保证： $n \leq 50$ ， $0 \leq a_i \leq 100$ 。
- 对于全部测试数据，保证： $2 \leq n \leq 1000$ ， $0 \leq a_i \leq 1000$ 。

翻滚的长方体 (cube)

【题目描述】

在一个 n 行 m 列的棋盘中，有一个长为 1、宽为 1、高为 2 的长方体机器人棋子竖立在棋盘的 $(1, 1)$ 位置上。机器人棋子可以在棋盘上放倒、竖直、翻滚，但是和普通长方体一样不能改变形状，问机器人棋子至少需要多少次操作，才能竖立在棋盘的 (n, m) 位置上。

【输入格式】

从文件 `cube.in` 中读入数据。

输入共一行，包含两个整数 n, m 。

【输出格式】

输出到文件 `cube.out` 中。

输出一个整数，表示这个长方体棋子需要最少多少次操作才能竖直放在棋盘的 (n, m) 位置上。若无解，输出 `impossible`。

【样例 1 输入】

1 1 1

【样例 1 输出】

1 0

【样例 2 输入】

1 2 4

【样例 2 输出】

1 3

【样例 3 输入】

1 3 7

【样例 3 输出】

1 6

【数据范围】

- 对于 20% 的数据, 保证: $1 \leq n \leq m \leq 10$ 。
- 对于 60% 的数据, 保证: $1 \leq n \leq m \leq 1000$ 。
- 对于全部测试数据, 保证: $1 \leq n \leq m \leq 10^9$ 。