

CSP-X2022 小学组二轮补赛试题（上半场）

考试时间：2023 年 3 月 4 日下午 14: 30--16: 00

题目名称	独木桥	移动棋子
题目类型	传统型	传统型
目录	bridge	chess
程序名称	bridge.cpp	chess.cpp
输入文件名	bridge.in	chess.in
输出文件名	bridge.out	chess.out
测试点数量	10	10
每测试点时限	1 秒	1 秒
每测试点分值	10	10
内存限制	512M	512M

注意事项

1、代码必须放在子文件夹内，子文件夹名与题目英文名一致。文件名（包括程序名和输入输出文件名）必须使用英文小写。

2、C++编译选项：-O2 -std=c++14。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

3、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。

第一题 独木桥 (bridge)

题目描述

长度为 L 米的独木桥上有 n 个人，他们每个人都想以最快的时间离开危险的独木桥。

已知每个人在独木桥上的行走速度为 1 米/秒，每个人只要能走到独木桥的两个端点中的其中一个就可以离开独木桥。

由于独木桥的桥面宽度很窄，只能容纳一个人通过，当两个人相遇时，他们无法交错通过，只能各自调转方向，继续沿反方向行走。

给你独木桥上的人数 n ，独木桥的长度 L ，第 i 个人的初始位置到独木桥左端点的距离 a_i 米（每个人开始的朝向未知，但他们可以根据需要随时调转行走的方向）。

请计算出所有人同时出发，全部都离开独木桥所需的最短时间。

输入格式（输入文件为 bridge.in）

第一行一个整数 n ，表示人数。

第二行一个整数 L ，表示独木桥的长度（米）。

第三行是 a_1, a_2, \dots, a_n ，其中 a_i 表示第 i 个人初始位置到独木桥左端点的距离。

输出格式（输出文件为 bridge.out）

输出一行一个整数，表示所有人都离开独木桥所需的最短时间。

输入输出样例

样例 1 输入

```
3
10
2 6 7
```

样例 1 输出

```
4
```

说明：三个人同时出发，第一个人向左走，需要 2 秒离开桥，第二个人向右走需要 4 秒离开桥，第三个人向右走需要 3 秒离开桥。所以，4 秒后，三个人都离开了独木桥。

样例 2 输入

```
7
214
11 12 7 13 176 23 191
```

样例 2 输出

```
38
```

数据范围:

- 对于 50% 的数据: $1 \leq n \leq 10^3$;
- 对于 100% 的数据: $1 \leq n \leq 10^6$, $1 \leq L \leq 10^6$, $0 \leq a_i \leq L$ 。

第二题 移动棋子 (chess)

题目描述

一维的棋盘上有无限多个格子, 每个格子都有一个编号, 最中间的格子编号为 0, 0 号格子向右依次编号为 1, 2, 3, ..., 向左依次编号为 -1, -2, -3, ...。

小明的目标是要将一枚棋子从 x 号格子移动到 y 号格子。

每一次操作有两种选择:

操作 1: 向右移动 1 个格子。

操作 2: 从当前棋子所在的 a 号格子, 直接跳到 $-a$ 号格子 (如: 可以从 6 直接跳到 -6, 也可以从 -6 直接跳到 6)。

可以证明, 无论整数 x 和 y 的值是多少, 目标总是可以实现的。

请你设计程序, 帮小明计算把棋子从 x 号格子移动到 y 号格子需要的最少操作次数。

输入格式 (输入文件为 chess.in)

一行, 两个整数 x 和 y , 表示要将棋子从 x 号格子移动到 y 号格子。

输出格式 (输出文件为 chess.out)

一个整数, 表示小明把棋子从 x 号格子移动到 y 号格子需要的最少操作次数。

输入输出样例

样例 1 输入

10 20

样例 1 输出

10

样例 1 说明：需要进行 10 次操作 1。

样例 2 输入

10 -10

样例 2 输出

1

样例 2 说明：需要进行 1 次操作 2。

样例 3 输入

-10 -20

样例 3 输出

12

样例 3 说明：先进行 1 次操作 2，然后进行 10 次操作 1，最后进行 1 次操作 2。

数据范围

- 30% 的数据： $|x| \leq 100, |y| \leq 100$ 。
- 100% 的数据： x, y 都为整数； $|x| \leq 10^9, |y| \leq 10^9$ ； x, y 不相等。
- 有 40% 的数据， $|x| > |y|$ 。
- 有 40% 的数据， $|x| < |y|$ 。
- 有 20% 的数据， $|x| = |y|$ 。