

2024 西安市信息学算法编程大赛

小低组复赛

时间: 2024 年 4 月 27 日 09:00 ~ 12:00

题目名称	买书	可行时间段	交替字符串	倒豆子
题目类型	传统型	传统型	传统型	传统型
目录	book	time	alternate	bean
可执行文件名	book	time	alternate	bean
输入文件名	book.in	time.in	alternate.in	bean.in
输出文件名	book.out	time.out	alternate.out	bean.out
每个测试点时限	1.0 秒	1.0 秒	1.0 秒	1.0 秒
内存限制	512 MiB	512 MiB	512 MiB	512 MiB
测试点题目	10	10	10	10
测试点是否等分	是	是	是	是

提交源文件程序名

book.cpp	time.cpp	alternate.cpp	bean.cpp
----------	----------	---------------	----------

编译选项

-O2 -std=c++14

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. C/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. 提交的程序代码文件的放置位置请参考考场要求。
4. 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。
5. 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）。
6. 选手提交的程序源文件必须不大于 100 KB。
7. 程序可使用的栈空间内存限制与题目的内存限制一致。

买书 (book)

【题目描述】

小 C 非常喜欢学习信息学，老师给他推荐了两本书参考学习。

两本书所在书店与小 c 家在一条直线上，把直线看作一个数轴，那么两本书所在书店和小 c 的位置坐标分别是 a, b, s 。

小 c 准备走路去买书，他每秒可以沿着直线正方向或者负方向走一米。小 c 想知道最少需要花多少秒才能买到两本书。(假设掉头不花费时间)

【输入格式】

从文件 `book.in` 中读入数据。

输入共一行，三个整数 a, b, s 。

【输出格式】

输出到文件 `book.out` 中。

输出共一行，一个整数，表示小 c 买到两本书至少需要多少秒。

【样例 1 输入】

```
1 -1 0
```

【样例 1 输出】

```
3
```

【样例 2 输入】

```
-2 -2 217
```

【样例 2 输出】

```
219
```

【样例 1 解释】

小 C 可以先沿着数轴的负方向走 1 秒，再沿着数轴的正方向走 2 秒。

可以证明小 C 至少需要走 3 秒。

【样例 2 解释】

小 C 可以直接沿着数轴的负方向走 219 秒。

可以证明小 C 至少需要走 219 秒。

【数据范围】

对于 100% 的数据， $|a|, |b| \leq 500$ 。

可行时间段 (time)

【题目描述】

一天有 24 个小时，据专家建议，对小学生来说当日 23 到次日 7 点是最佳睡眠时间，由于入眠也需要时间，因此最好 22 点可以躺在床上。

另外，小学生每周一到周五 7 点到 17 点需要在学校上学。

所以对一个小学生来说，周一到周五可以完成课后作业的时间段预估为 17 点到 22 点，周六周日可以完成课后作业的时间段为 7 点到 22 点，不考虑从学校到家时间。

假设当前时间是周六 9 点，AI 老师计划让学生在 n 时以后开始做题，请你判断该时间段是否适合让学生完成课后作业。（学生完成作业需要一小时）

【输入格式】

从文件 `time.in` 中读入数据。

一个整数 n ，表示 AI 老师给学生安排在 n 时后开始答题。

【输出格式】

输出到文件 `time.out` 中。

输出共两行。

第一行为一个字符串，“yes”或者“no”，代表该时间段是否适合学生答题。

第二行两个整数，表示这个时间段，分为开始时间和结束时间。

【样例 1 输入】

10

【样例 1 输出】

yes
19 20

【样例 2 输入】

13

【样例 2 输出】

no
22 23

【数据范围】

对于 30% 的数据， $1 \leq n \leq 10^3$ 。

对于 100% 的数据， $1 \leq n \leq 10^8$ 。

交替字符串 (alternate)

【题目描述】

给定一个只包含两种字符的字符串 s ，如果 s 的一个子串中不存在两个相邻的字符相同的情况，则认为这是一个交替子字符串。

需要注意的是，两个子串起始位置不同，终止位置不同，则认为他们是不同的子串。

求字符串 s 的交替子字符串的个数。

【输入格式】

从文件 `alternate.in` 中读入数据。

一行，一个字符串 s ，仅包含两种字符。

【输出格式】

输出到文件 `alternate.out` 中。

一个整数，表示字符串 s 的交替子字符串的个数。

【样例 1 输入】

abbb

【样例 1 输出】

5

【样例解释 1】

共有“a”，“ab”，“b”，“b”，“b”五种不同的交替子串

【数据范围】

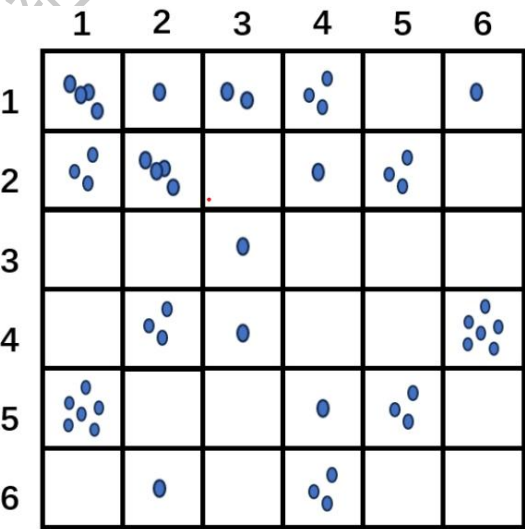
对于 30% 的数据， $1 \leq s.size() \leq 10^3$ 。

对于 100% 的数据， $1 \leq s.size() \leq 10^6$ 。

保证输入数据只出现大写字母和小写字母且字符串中有且仅有两种字符。

倒豆子 (bean)

【题目描述】



如图有一个正方形置物盘，置物盘上有 $n \times n$ 个小格子，每个小格子里有数量不等的豆子。对于整个置物盘，我们可以做以下两种操作之一：

操作一：选择置物盘中一行小格子，将这一行小格子中的豆子倒到相邻行中；再选择置物盘中一列小格子，将这一列小格子中的豆子倒到相邻列中。

操作二：将整个置物盘沿顺时针转动一圈。

假设需要做 m 次操作，请你选择最佳的操作过程，使得做完操作后，将其中的一个格子中的豆子拿出来作为种子时，种子的数量最多。

【输入格式】

从文件 `bean.in` 中读入数据。
总共 $n+2$ 行。
第一行一个正整数 n ，代表置物盘上的行数和列数。
第二到 $n+1$ 行每行 n 个非负整数 a_{ij} ，代表置物盘上第 i 行第 j 列的小格子里有 a_{ij} 个豆子。
第 $n+2$ 行一个正整数 m ，代表操作的次数。

【输出格式】

输出到文件 `bean.out` 中。
一个正整数，代表最终拿出来的豆子的数量。

【样例 1 输入】

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

【样例 1 输出】

```
10
```

【样例 2 输入】

```
6
4 1 2 3 0 1
3 4 0 1 3 0
0 0 1 0 0 0
0 3 1 0 0 6
6 0 0 1 3 0
0 1 0 3 0 0
1
```

【样例 2 输出】

```
12
```

【样例解释 1】

将第 1 行的豆子倒到第 2 行，再将第 1 列的豆子倒到第 2 列，此时第 2 行第 2 列的豆子数量是 10。

【样例解释 2】

（该样例如题目中的图片所示）将第 1 行的豆子倒到第 2 行，再将第 1 列的豆子倒到第 2 列，此时第 2 行第 2 列的豆子数量是 12。

【数据范围】

对于 20% 的数据， $2 \leq n \leq 10$, $1 \leq m \leq 10$, $0 \leq a_{i,j} \leq 100$ 。

对于 30% 的数据， $2 \leq n \leq 100$, $1 \leq m \leq 10$, $0 \leq a_{i,j} \leq 100$ 。

对于 100% 的数据， $2 \leq n \leq 10^3$, $1 \leq m \leq 10$, $0 \leq a_{i,j} \leq 10^8$ 。

特殊数据：对于 20% 的数据，保证 $m = 1$ 。