

2022 CCF 非专业级软件能力认证模拟赛 - 入门组

2022/7/31

(请选手务必仔细阅读本页内容)

题目名称	存钱	旅行	排序	异或
题目类型	传统型	传统型	传统型	传统型
英文题目名称	save	travel	sort	xor
输入文件名	save.in	travel.in	sort.in	xor.in
输出文件名	save.out	travel.out	sort.out	xor.out
输出文件名	1s	1s	1s	1s
内存上限	512M	512M	512M	512M
测试点数目	10	10	10	10
每个测试点分值	10	10	10	10
附加样例文件	无	有	有	有
结果比较方式	全文比较	全文比较	全文比较	全文比较

注意事项：

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()` 的返回类型必须是 `int`，程序正常结束时的返回值必须是 `0`。

1 存钱

(save.cpp/c)

【题目描述】

藤藤想存 n 元钱，他决定第 1 天存 1 元，第 2 天存 7 元，第 3 天存 49 元，以此类推，每天存的钱是前一天的 7 倍。

藤藤想知道几天后，存款的总额能大于等于 n 元钱。

【输入输出格式】

【输入格式】

从文件 save.in 中读入数据。

第一行输入一个整数 n ，表示藤藤想存的钱数。

【输出格式】

输出到文件 save.out 中。

一行一个整数表示答案。

【样例】

【样例 1 输入】

52

【样例 1 输出】

3

3 天后，藤藤的存款为 $1+7+49=57$ 元。

【数据范围与提示】

对于 40% 以内的数据， $1 \leq n \leq 10$ 。

对于 100% 以内的数据， $1 \leq n \leq 10000$ 。

2 旅行

(travel.cpp/c/pas)

【题目描述】

有一个地图，地图上存在 n 个小镇，小镇从 1 到 n 编号。第 i 个小镇的坐标为 (x_i, y_i) 。定义两个小镇的距离为曼哈顿距离，比如小镇 i 到小镇 j 的距离为 $|x_i - x_j| + |y_i - y_j|$ ， $|a|$ 表示取 a 的绝对值。

同时在我们发现有 m 个小镇建立了传送门，也就是说，可以在任何时候任何瞬间不花费任何代价，直接到达这 m 个小镇的任何一个。

你一开始在小镇 1，你想按 1 到 n 的顺序访问所有小镇按顺序做任务，请问需要走过的最短距离是多少。

【输入输出格式】

【输入格式】

输入文件 travel.in

第一行输入两个整数 n, m ，表示地图上小镇的数目和建立了传送门的小镇数量。

接下来 n 行，其中第 i 行输入两个整数 x_i, y_i 个小镇的坐标。

接下来一行输入 m 个整数，表示建立了传送门的小镇编号。

【输出格式】

输出文件 travel.out 只有一行一个整数表示答案

【样例】

【样例 1 输入】

```
6 2
-10 -10
10 10
0 0
1 -1 -1 1
2 1
3 6
```

【样例 1 输出】

```
21
```

一开始在小镇 1，完成任务后，传送到小镇 6。
步行至小镇 2，累计 17 点距离，完成任务后，传送到小镇 3。
在小镇 3 完成任务，累计 17 点距离。
步行到小镇 4 完成任务，累计 19 点距离，传送到小镇 3。
步行到小镇 5 完成任务，累计 21 点距离。
传送到小镇 6 并完成任务。
共计 21 点距离。

【数据范围与提示】

对于 20% 的据有 $m = 0$ 。
对于 40% 的据有 $m = 1$ 。
对于 60% 的数据有 $n, m \leq 300$
对于 100% 的数据有 $1 \leq n \leq 5000, 0 \leq m \leq n, -10000 \leq x_i, y_i \leq 10000$
数据保证没有任意两个小镇在同一坐标下。

3 排序

(sort.cpp/c/pas)

【题目描述】

给定长度为 n 的整数的序列 a_1, a_2, \dots, a_n , 你可以将其中任意两个整数相乘后得到新的序列 $b_1, b_2, \dots, b_{\frac{n \times (n-1)}{2}}$,

现在要求你计算出新序列从大到小排序后第 t 大的数字是多少。

【输入输出格式】

【输入格式】

从文件 sort.in 中读入数据

输入共两行:

第一行两个正整数 n, t

第二行 n 个整数表示 a_1, a_2, \dots, a_n

【输出格式】

输出到文件 sort.out

输出共一行一个整数表示答案

【样例】

【样例 1 输入】

4 5

7 -2 8 4

【样例 1 输出】

-14

结果共-14,56,28,-16,-8,32 这 6 个数, 其中第 5 大数字为-14

【数据范围与提示】

对于 30% 的数据, $1 \leq n \leq 500, -2 \times 10^9 \leq a_i \leq 2 \times 10^9$

对于另外 40% 的数据, $1 \leq n \leq 10^5, 1 \leq a_i \leq 2 \times 10^9$

对于 100% 的数据, $1 \leq n \leq 10^5, -2 \times 10^9 \leq a_i \leq 2 \times 10^9$

数据保证: $1 \leq t \leq \frac{n \times (n-1)}{2}$

4 异或

(xor.cpp/c/pas)

【题目描述】

给定一个长度为 k 的数列 x_1, x_2, \dots, x_k 作为一个起始值，然后从第 $k+1$ 项开始，无限地延展这个序列，延展后数列 x_i 的定义如下：

$$x_i = x_{i-1} \oplus x_{i-2} \oplus \dots \oplus x_{i-k}$$

这其中 \oplus 是将两个数字二进制编码后进行异或操作的意思。

给定 q 个询问，每个询问都带有一组参数 l, r ，请计算并输出

$x_l \oplus x_{l+1} \oplus \dots \oplus x_r$ 的值。

【输入输出格式】

【输入格式】

从文件 xor.in 中读入数据

第一行单个整数表示 k ；

第二行 k 个整数表示 x_1, x_2, \dots, x_k ；

第三行单个整数表示 q ；

第四行到第 $q+3$ 行：每行两个整数 l_i 与 r_i ，表示对一个区间的询问。

【输出格式】

输出到文件 xor.out 中

共 q 行对每个询问，输出一个整数表示答案。

【样例】

【样例 1 输入】

```
4
1 3 5 7
3
2 2
2 5
1 5
```

【样例 1 输出】

```
3
1
0
```

【样例 2 输入】

```
5
3 3 4 3 2
4
1 2
1 3
5 6
7 9
```

【样例 2 输出】

```
0
4
7
4
```

【数据范围与提示】

$0 \leq x_i \leq 10^9$;

记查询参数 l_i 与 r_i 的最大值为 n 。

对于 30% 的分数, $k \leq 100$, $n \leq 1000$, $q \leq 1000$;

对于 60% 的分数, $n \leq 10^5$

对于 100% 的分数, $k \leq 10^5$, $n \leq 10^9$, $q \leq 10^5$