

2024 洛谷网校秋令营普及组 模拟赛

2024 衔接计划模拟赛 5

时间：2024 年 10 月 4 日 12:00 ~ 08:30

题目名称	代码	三七	小朋友	异或序列
题目类型	传统型	传统型	传统型	传统型
目录	cppcode	one	xpy	sequence
可执行文件名	cppcode	one	xpy	sequence
输入文件名	cppcode.in	one.in	xpy.in	sequence.in
输出文件名	cppcode.out	one.out	xpy.out	sequence.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
测试点数目	10	10	10	10
测试点是否等分	是	是	是	是

提交源程序文件名

对于 C++ 语言	cppcode.cpp	one.cpp	xpy.cpp	sequence.cpp
-----------	-------------	---------	---------	--------------

编译选项

对于 C++ 语言	-O2 -std=c++14 -static
-----------	------------------------

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本省的具体要求。
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 100KB。
7. 程序可使用的栈空间内存限制与题目内存限制一致。
8. 只提供 Linux 格式附加样例文件。
9. 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。

代码 (cppcode)

【题目描述】

Z 最近在看小朋友们的代码，但是小朋友们的代码写的并不规范好看，现在请你帮 Z 规范化以下小朋友们的代码。

给出一个小朋友写的 n 行的代码，你需要对这 n 行代码做以下处理：

1. 去除掉代码中的空行，空行定义为由不可见字符组成的行。
2. 去除掉代码中连续的多个空格，将其修改为一个空格。

字符分为可见字符和不可见字符，其中可见字符指的是字母、数字、ASCII 字符中的可见符号，不是可见字符的字符为不可见字符。

请你输出处理之后的代码。

【输入格式】

从文件 *cppcode.in* 中读入数据。

第一行一个正整数 n ，表示代码的行数。

接下来 n 行，每行一个字符串，依次表示了小朋友代码的内容。

【输出格式】

输出到文件 *cppcode.out* 中。

输出若干行，表示经过你处理之后的小朋友代码内容。

【样例 1 输入】

```
1 13
2 #include<bits/stdc++.h>
3 using namespace std;
4 int main()
5 {
6
7
8     string s;
9     int n;
10    cin>>n;
11    getline(cin,s);
12    for(int i=1;i<=n;i++)
13        getline(cin,s);
```

14 }

【样例 1 输出】

```
1 #include<bits/stdc++.h>
2 using namespace std;
3 int main()
4 {
5     string s;
6     int n;
7     cin>>n;
8     getline(cin,s);
9     for(int i=1;i<=n;i++)
10     getline(cin,s);
11 }
```

【提示】

对于前 20% 的数据，满足 $n = 1$ ；

对于前 40% 的数据，保证代码中没有空行；

对于另外 10% 的数据，保证代码中没有空格；

对于全部的测试数据，满足 $1 \leq n \leq 1000$ ，代码中只包括大小写英文字母，ASCII 符号和空格以及换行符，每行字符总数不超过 100。

三七 (one)

【题目描述】

Z 获得了 T 个名叫三七的密码锁，为了打开这些密码锁，Z 必须解开密码锁上写着的一列数字的谜底。

每个密码锁上写了一列 n 个正整数 $a_1, a_2, a_3 \cdots a_n$ ，经过 Z 的观察，Z 每次操作可以在这个序列中选择两个位置 i, j ，并且将 a_i 替换为 $a_i \bmod a_j (a_j \neq 0)$ 。Z **只能并且必须** 进行这样的操作 3 次或者 7 次，密码锁的密码等于 Z 完成了操作之后序列中所有数字的 **和** 的最大值。

请你帮 Z 解开这些三七密码锁。

【输入格式】

从文件 `one.in` 中读入数据。

第一行一个整数 T ，表示 Z 获得的密码锁个数。

接下来 $2T$ 行，每两行描述一个密码锁的内容，其中的第一行输入一个整数 n ，表示密码锁上的正整数个数，其中的第二行依次输入 n 个正整数 $a_1, a_2, a_3 \cdots a_n$ 。

【输出格式】

输出到文件 `one.out` 中。

输出 T 行每行一个整数，表示这个密码锁的密码。

【样例 1 输入】

```
1 1
2 10
3 74 87 23 33 22 50 16 10 68 66
```

【样例 1 输出】

```
1 449
```

【样例 2】

见选手目录下的 `one/one2.in` 与 `one/one2.ans`。

【提示】

对于前 10% 的数据，满足 $n = 2$ ，并且 $a_1 \neq a_2$ ；

对于另外 30% 的数据，满足 $n \leq 10$ ，并且最优答案一定可以通过三次操作达成；

对于全部的测试数据，满足 $2 \leq n \leq 10^5, 1 \leq a_i \leq 10^9, 1 \leq T \leq 10$ 。

小朋友 (xpy)

【题目描述】

体育课上，有 n 个小朋友站成一排，小朋友的年龄在 $[11, 22]$ 岁之间。现在作为老师，你想把小朋友划分为若干组来进行游戏。在一个合法的分组方案中，你可以把相邻的小朋友分在一组，或者是一个小朋友独自一组。

特别的，老师不希望任意两组小朋友的小组人数和年龄完全一致，这样可以增加游戏的趣味性。

现在请你求出合法的分组方案数对 998244353 取模的结果。

【输入格式】

从文件 `xpy.in` 中读入数据。

第一行一个整数 n ，表示小朋友个数。

第二行 n 个整数，依次表示从左到右第 i 个小朋友的年龄 a_i 。

【输出格式】

输出到文件 `xpy.out` 中。

一行一个整数，表示答案。

【样例 1 输入】

```
1 6
2 21 12 18 16 16 21
```

【样例 1 输出】

```
1 7
```

【样例 2 输入】

```
1 5
2 21 15 17 19 15
```

【样例 2 输出】

```
1 6
```

【提示】

对于 20% 的数据，满足 $n \leq 10$ 。

对于 40% 的数据，满足 $n \leq 20$ 。

对于 60% 的数据，满足 $n \leq 40$ 。

对于 100% 的数据，满足 $1 \leq n \leq 60, 11 \leq a_i \leq 22$ 。

异或序列 (sequence)

【题目描述】

有一个长度为 n 的序列 a ，序列中的每个值在 $[0, 1024]$ 之间。请你求出这个序列有多少对连续子序列 (A, B) ，满足 A 在 B 之前，且 A, B 中所有元素的异或和为 m 。

形式化的，你需要求出有多少个四元组 (l_1, r_1, l_2, r_2) ，满足 $l_1 \leq r_1 < l_2 \leq r_2, (\oplus_{i=l_1}^{r_1} a_i) \oplus (\oplus_{i=l_2}^{r_2} a_i) = m$ 。

【输入格式】

从文件 `sequence.in` 中读入数据。

第一行两个整数 n, m ，表示数组长度，异或和。

第二行 n 个整数，表示数组 a 。

【输出格式】

输出到文件 `sequence.out` 中。

一行一个整数，表示答案。

保证答案不超过 `long long` 表示范围。

【样例 1 输入】

```
1 5 3
2 255 667 2 687 147
```

【样例 1 输出】

```
1 0
```

【样例 2 输入】

```
1 10 3
2 722 324 110 891 150 317 791 587 117 933
```

【样例 2 输出】

```
1 1
```


【样例 3】

见选手目录下的 *sequence/sequence3.in* 与 *sequence/sequence3.ans*。

【提示】

对于 20% 的数据，满足 $n \leq 20$ 。

对于 40% 的数据，满足 $n \leq 50$ 。

对于 60% 的数据，满足 $n \leq 200$ 。

对于 80% 的数据，满足 $n \leq 5000$ 。

对于 100% 的数据，满足 $1 \leq n \leq 100000, 0 \leq a \leq 1024, 0 \leq m \leq 1024$ 。