

ZZYZ Round #3. Unrated

郑州一中的蒟蒻出题组

测试时间: 4.5小时.

一. 题目概览

中文题目名称	紫水晶教练维尼	紫水晶教练之争	柠檬树	欧拉
英文题目与子目录名	weini	amethyst	lemon	euler
可执行文件名	weini	amethyst	lemon	euler
输入文件名	weini.in	amethyst.in	lemon.in	euler.in
输出文件名	weini.out	amethyst.out	lemon.out	euler.out
每个测试点时限	1 秒	1 秒	1 秒	2 秒
测试点数目	100	10	20	10
每个测试点分值	1	10	5	10
附加样例文件	无	无	无	无
结果比较方式	全文比较 (过滤行末空格及文末回车)			
题目类型	传统	传统	传统	传统
运行内存上限	128M	128M	128M	128M
题目特点&建议	快抢一血	简单模拟	通俗易懂	小学数学

二. 提交源文件程序名

对于 C++ 语言	weini.cpp	amethyst.cpp	lemon.cpp	euler.cpp
对于 C 语言	weini.c	amethyst.c	lemon.c	euler.c
对于 pascal 语言	weini.pas	amethyst.pas	lemon.pas	euler.pas

三. 编译命令 (不包含任何优化开关)

对于 C++ 语言	g++ -o weini.cpp -lm	g++ -o amethyst.cpp -lm	g++ -o lemon.cpp -lm	g++ -o euler.cpp -lm
对于 C 语言	gcc -o weini.c -lm	gcc -o amethyst.c -lm	gcc -o lemon.c -lm	gcc -o euler.c -lm
对于 pascal 语言	fpc weini.pas	fpc amethyst.pas	fpc lemon.pas	fpc euler.pas

注意事项:

0. 如果在洛谷上进行评测,那么1~9条无效.
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不需要建立额外的子文件夹.
4. 因违反以上三点而出现的错误或问题, 不能重新评测.
5. 程序可使用的栈内存空间限制与题目的内存限制一致.
6. 评测时采用的机器配置为: Intel(R) Core(TM) i7-8700K CPU @ 3.70GHz, 内存 8 GB 上述时限以此配置为准.
7. 只提供 Windows 格式附加样例文件.(机房没有Linux).
8. 评测在Star的电脑上进行, 各语言的编译器版本以其为准.
9. 最终评测时所用的编译命令中不含任何优化开关.
10. 请不要直接复制该PDF中的样例. 对应样例请前往对应的选手文件夹中获取.

记号与说明:

- 累加记号: $\sum_{i=l}^r a_i$ 表示 $a_l + a_{l+1} + \cdots + a_r$. 如果只有下标, 则对于所有满足下标的内容求和.
- 累异或记号: $\bigoplus_{i=l}^r a_i$ 表示 $a_l \oplus a_{l+1} \oplus \cdots \oplus a_r$. 如果只有下标, 则对于所有满足下标的内容异或.
- 区间max记号: 表示满足下标条件的所有值中的最大值
- \forall 记号: 意为任意.
- 对于本次考试而言 $[x, y](x < y)$ 记号: 表示 x 到 y 中所有的整数组成的集合(包括 x, y).
- $a \in S$ 记号: 表示 a 元素属于集合 S .
- 定义一个数列 b_1, b_2, \dots, b_n 为回文数列当且仅当该数列满足 $\forall i \in [0, n-1], b_{l+i} = b_{r-i}$.
- 关于hack的背景介绍: 在比赛中, 当某道题通过之后, 查看其他选手该题的代码时试图找出其中的漏洞, 自己造一个数据让这个代码不能通过, 这就是Hack.

1 紫水晶教练维尼〔weini.cpp/c/pas〕

1.1 题目背景

紫水晶维尼教练是一个非常强的OIer,尤其擅长解决现实问题.

1.2 题目描述

这天他有点饿,想吃一些苹果,一共有 n 个苹果,编号从1到 n .每一个苹果都有自己的饱食度,紫水晶教练维尼可以吃任意多的苹果,但是要求吃的所有的苹果的序号都是连续的,可以获得的总饱食度是吃的苹果的饱食度之和乘以吃的苹果的个数,也就是

$$\max_{1 \leq l \leq r \leq n} \left(\sum_{l \leq i \leq r} a_i \right) (\text{保证饱食度是整数})$$

虽然他十分的饿,但是他毫不费力地想出了这样的算法,以伪代码描述:

```
int solve(int n, int a[]){
    // a[] is an array with length n that may be dealt like an array.
    // Number starts from one.
    int re=0, now=0, k=0;
    for(int i = 1;i<=n;i++){
        now = now+a[i];
        if(now<0){
            now = 0;
            k = i;
        }
        re = max(re, (i-k)*now);
    }
    return re;
}
```

但是聪明的你一定发现了,紫水晶教练维尼写的是不正确的. 比如对于这组数据: 2 -5 100000, 按照教练维尼的代码结果是 1×100000 ,但正确答案应该是 $(2 - 5 + 100000) \times 3$.

于是你准备拿一组**hack**数据来嘲笑紫水晶教练维尼,为了使嘲笑更有力度,你拿出的数据一定要满足这样的一些条件.

- 正确答案-维尼的答案= k
- **hack**数据的 n 最小,即序列尽量短.
- 要求**hack**数据的字典序最小, 也就是 $|a_1|$ 最小,在 $|a_1|$ 相等时要求 $|a_2|$ 尽量小,在 $|a_1|$ 和 $|a_2|$ 均相等时要求 $|a_3|$ 最小,以此类推.
- 要求 a_1 是一个负整数.

我们可以证明在如上条件的限制下,一定存在唯一解.

1.3 输入格式

输入文件的第一行包含一个整数 T ,表示数据的组数
接下来分别给出 T 组数据,每组数据的第一行包含一个 k .

1.4 输出格式

输出文件共有 T 行,对于每组数据,首先输出一个正整数 n 表示你构造的 $hack$ 数据的长度,接下来在同一行输出 $a_1, a_2, a_3, \dots, a_n$.

1.5 输入输出样例

weini.in	weini.out
1	2 -1 3
1	

1.6 说明/提示

数据范围:

- 对于10%的数据,保证 $1 \leq T \leq 10, 1 \leq k \leq 10$.
- 对于40%的数据,保证 $1 \leq T \leq 100, 1 \leq k \leq 2^{31} - 1$.
- 对于100%的数据,保证 $1 \leq T \leq 1000, 1 \leq k \leq 2^{63} - 1$.

2 紫水晶教练之争〔amethyst.cpp/c/pas〕

2.1 题目背景

lyfx因为不服Kingxbz紫水晶教练的名头，于是信心十足地前来挑战。

Kingxbz不屑于和这种黑铁教练公平竞争，于是让lyfx先来选比赛形式。“没有人比我更懂斐波那契”，lyfx想用斐波那契数列拿下一胜，可是kingxbz也不甘示弱。出了一道看似公平的规则。可惜的是来的太仓促没有准备东西，双方决定切题来一决雌雄。Kingxbz和lyfx叫上蒟蒻Vinnie从CODE FORCES上下载了两种题目，共进行t场比赛，每场比赛开始时，两个人轮流按照双方的规则开始切题。

2.2 题目描述

按照lyfx和kingxbz的规则：

- lyfx的规则:每个人先选择斐波那契数列中的一个数，任选一堆然后切相应数量的题.如(1,2,3,5,8...)
- kingxbz的规则:每个人选择一种题切若干数量个，然后可以选择是否继续切同样的数量的另一种题。

两人轮流开始切以便互相监督，谁最后切完最后一题谁就获胜，每局开始时lyfx都毫不客气的占据了先手,他们比着比着发现题实在太多了，虽然两个人已经用最快速度切题了，但是比赛进行的还是很慢，以至于在一旁观战的BLUESKY007巨佬忍不住说道：‘不用比了，我知道谁会赢’。

蒟蒻Vinnie看的一头雾水，你能帮帮他求出谁一定能赢吗？

2.3 输入格式

第一行给定整数t，表示比赛场数。

奇数场用lyfx的规则，而偶数场用Kingxbz的规则。

接下来t行，每行两个整数a,b，表示两种题目的数量。

2.4 输出格式

共t行，每行输出一个字符串：

如果这场比赛lyfx赢了，输出lyfx yyds!

如果这场比赛Kingxbz赢了，输出xbz yyds!

2.5 输入输出样例

amethyst.in	amethyst.out
2	
1 1	xbz yyds!
1 4	lyfx yyds!

2.6 说明/提示

对于其中20%的数据, $a = b$.

对于其中10%的数据, $t = 2$.

对于其中另外10%的数据, $a, b \leq 10$.

对于100%的数据, $t \leq 10; 1 \leq a, b, c \leq 1000$.

3 柠檬树 [lemon.cpp/c/pas]

3.1 题目背景

Yuzaki Nasa 走出家门, 发现了一棵巨大的柠檬树. 这棵柠檬树以1为根节点, 共有 n 个节点. 他准备用这棵树研究问题.

3.2 题目描述

Yuzaki Nasa非常在意一个数列是不是「NASA」序列. 「NASA」序列的定义是这样的: 数列中不存在长度大于1且长度为偶数的回文子串.

例 1 比如, 序列 $\{2, 3, 4, 5, 6\}$ 是「NASA」序列. 因为不存在回文子串.

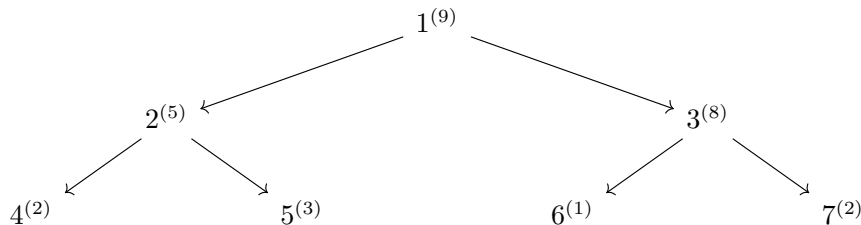
然而序列 $\{1, 2, 3, 4, 4, 3, 5\}$ 不是「NASA」序列. 因为存在 $[3, 4, 4, 3]$ 这个长度为4的回文子串.

这棵树的每个节点上面都有一个数. 这个数本应在 $[1, k]$ 之间. 但由于柠檬生长过程中柠檬酸堆积, 有一些节点上的数受到了腐蚀变成了0.

Yuzaki Nasa从节点编号1行走至节点编号 x . 他所经过的简单路径上面会经过一些节点. 这些节点上面的数字按照行走顺序排成一行构成一个数列.

Yuzaki Nasa想知道, 这个数列在被腐蚀之前有多少种可能是「NASA」序列. 假设这个答案是 h . 那么定义 $G(x) = h \bmod 998244353$.

例 2 考虑如下一棵树: 中间的数字表示树的节点编号, 右上角的数字表示该节点的数字.



比如这棵树中节点编号从1到5的简单路径需要经过1, 2, 5号节点. 1, 2, 5号节点上面的数字为9, 5, 3. 我们排出来的序列为 $[9, 5, 3]$.

由于序列 $[9, 5, 3]$ 不存在回文子串, 因此这是一个「NASA」序列. 由于所有上的数字大于0, 说明没有节点被腐蚀. 所以原来的「NASA」序列只有一种情况. 也就是 $G(5) = 1 \bmod 998244353 = 1$.

Yuzaki Tsukasa认为这个问题有点简单. 所以她想知道 $G(1), G(2), G(3), \dots, G(n)$ 的答案, 并且把它们异或起来并输出最终答案. 也就是输出:

$$\bigoplus_{i=1}^n G(i).$$

的答案.

3.3 输入格式

第一行, 输入两个数 n, k . 表示这个树有 n 个节点, 可以从 $[1, k]$ 中选择被腐蚀掉的数.

第二行, n 个正整数 a_1, a_2, \dots, a_n . 其中 $\forall i \in [1, n], a_i \in [0, n]$. 其中, 0表示该节点被腐蚀, 否则该权值就是确定的.

第三行, $(n-1)$ 个正整数, 第 i 个数表示第 $i+1$ 个节点的父亲.

3.4 输出格式

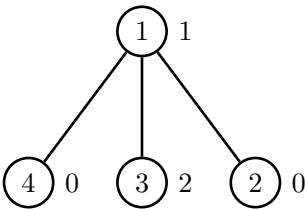
共一行,其中第一行输出 $\bigoplus_{i=1}^n G(i)$ 的结果.

3.5 输入输出样例

lemon.in	lemon.out
4 4 1 0 2 0 1 1 1	0

3.6 说明/提示

对于样例1的解释:
树的形态如下图.



其中, 圆圈右侧的节点表示该节点权值, 圆圈内部节点表示该节点的编号.那么:

- 从1到1, NASA序列共有一种情况. 也就是 $G(1) = 1 \bmod 998244353 = 1$.
- 从1到2, 注意到第二号节点可以填入2或3或4. 也就是 $G(2) = 3 \bmod 998244353 = 3$.
- 从1到3, NASA序列共有一种情况. 也就是 $G(3) = 1 \bmod 998244353 = 1$.
- 从1到4, 同理, $G(4) = 3 \bmod 998244353 = 3$.

那么要求输出答案即为: $1 \oplus 1 \oplus 3 \oplus 3 = 0$.

对于样例2的解释:我有一个绝妙的解释, 可惜地方太小, 写不下.
数据范围

- 对于10%的数据, 保证是一条链, $n \leq 20, k \leq 5$, 0的个数少于10个.
- 对于50%的数据, 保证是一条链, $n \leq 1005, k \leq 1005$.
- 对于100%的数据, $n \leq 10^5, k \leq 10^5$. $a_i \in [0, n]$. 保证给定的非零数列中不存在NASA序列.

4 欧拉〔euler.cpp/c/pas〕

4.1 题目背景

除非你能A了这道题，不然你可能永远不会知道这题为什么叫Euler.

-by kingxbz

金牌教练Star让小蒟蒻Kingxbz出题，于是Kingxbz绞尽脑汁，冥思苦想，终于想出了一道题：

试题1.1 给定两个正整数a,b($0 < a, b < 100$)，求a+b的值.

但是他不会写std，于是就找同机房大佬TopTom帮忙.TopTom觉得单纯的加法太简单了，随手就把问题改成了

试题1.2 给定两个正整数列a,b($0 < a, b < 100$)，求 $\sum_{i=1}^n \sum_{j=1}^n a_i b_j$ 的值.

但是TopTom觉得Kingxbz太菜了，不给他写std.于是Kingxbz就去找lyfx. lyfx觉得还是太简单了，就悄悄的修改了一下：

试题1.3 给定两个正整数列a,b($0 < a, b < 100$)，求 $\sum_{i=1}^n \sum_{j=1}^n (a_i b_j - a_j b_i)^2$ 的值.

但是lyfx觉得Kingxbz太菜了，不肯告诉他怎么写，于是Kingxbz只得去找AK巨兽BLUESKY007. BLUESKY007一看，数据范围辣么小也叫题？反手就改成了：

试题1.4 给定两个正整数列a,b($0 < a, b < 10^6$)，求 $\sum_{i=1}^n \sum_{j=1}^n (a_i b_j - a_j b_i)^2$ 的值

BLUESKY007告诉Kingxbz这题有手就行，但是Kingxbz还是不会写，于是就找金牌教练Star. Star告诉Kingxbz 数学题不配上数据结构那能叫题？反手把题改成了：

4.2 题目描述

给定两个正整数列a,b($0 < a, b < 10^6$)

有下面几个操作：

- 操作1：给定 i, x, y ，把 a_i, b_i 改成 x, y ，其中 $(x, y \leq 10^9)$ 。
- 操作2：求 $\sum_{i=l_1}^{r_1} \sum_{j=l_1}^{r_1} (a_i b_j - a_j b_i)^2$ 和 $\sum_{i=l_2}^{r_2} \sum_{j=l_2}^{r_2} (a_i b_j - a_j b_i)^2$ 的值分别模 $10^9 + 7$ 后的值。

操作2的求和操作求出的数据太大了，小蒟蒻Kingxbz看不懂，所以为了降低题目难度，让所有人轻松AC，你只需输出 $\gcd(p^a - 1, p^b - 1)$ 模998244353的值就好了。

其中：

$$a = \sum_{i=l_1}^{r_1} \sum_{j=l_1}^{r_1} (a_i b_j - a_j b_i)^2 \mod 1 \times 10^9 + 7$$

$$b = \sum_{i=l_2}^{r_2} \sum_{j=l_2}^{r_2} (a_i b_j - a_j b_i)^2 \mod 1 \times 10^9 + 7$$

可惜不管怎样，Kingxbz还是不会写，希望你能来帮这只蒟蒻解决这些问题。

4.3 输入格式

第一行给定两个正整数 n, q
接下来两行，每行给定 n 个整数，分别为 a_i, b_i
接下来 q 行，每行一个1或者2表示操作种类
若是1，紧跟着三个整数 i, x, y 表示修改操作
若是2，紧跟着五个整数 p, l_1, r_1, l_2, r_2 表示查询操作

4.4 输出格式

对于每次询问，输出取模后的gcd值，中间输出换行.

4.5 输入输出样例

euler.in	euler.out
3 3	
1 2 3	
3 2 1	159509353
2 3 1 3 1 2	3
1 2 6 1	
2 2 1 3 2 3	

4.6 说明/提示

对于20%的数据, $n, w \leq 10^3, a_i, b_i, x, y \leq 10^3, p \leq 10$
对于100%的数据, $n, q \leq 10^6, p, a_i, b_i, x, y, \leq 10^9$