2023 (ICPC) Jiangxi Provincial Contest

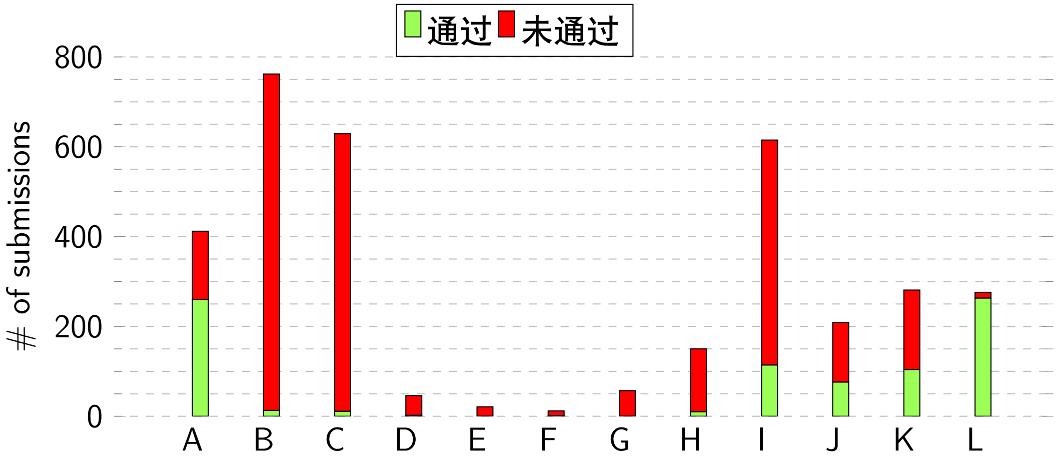
WHU ICPC 命题组

May. 21th, 2023

比赛小结

- 本次比赛共收到 3550 份提交代码。
- 其中 860 份代码正确。
- 263 个参赛队伍有提交记录。
- 所有参赛队伍至少通过一题。
- 12 道题中的 10 题有队伍通过。

各题通过情况



A. Drill Wood to Make Fire

题意

签到

题解

输出 $S \times V \geqslant N$

B. Wonderful Array

题意

先给 k 个数 $a_0, a_1, \dots, a_{k-1}$ ，再给三个数 n, m, x

令 $b_i = \begin{cases} x, & i = 0 \\ b_{i-1} + a_{(i-1) \bmod k}, & 0 < i \leqslant n \end{cases}$

求 $(b_i \bmod m) \leqslant (b_{i+1} \bmod m)$ 的个数，其中 $(0 \leqslant i < n)$

题解

发现 $=$ 的个数就是看有多少 $a_i \bmod m = 0$，$<$ 的个数不好求，那么就考虑求 $>$ 的个数。

先将所有 a_i 对 m 取模，不会影响答案。因为 $0 \leqslant b_{i+1} - b_i < m$

$$b_i \bmod m > b_{i+1} \bmod m \iff \lfloor b_i/m \rfloor < \lfloor b_{i+1}/m \rfloor$$

整体考虑的话就会发现，$>$ 的个数就是 $\lfloor \frac{b_n}{m} \rfloor$

C. Battle

题意

博弈，取石子游戏，但只能取一个指定数 p 的幂（包括 p^0）

C. Battle

题解

对于有多组的博弈，考虑采用 sg 函数，通过异或合并。

第一种方式是打表找规律，通过打表可以发现

- 当 p 为奇数时

$$sg(x) = \begin{cases} 0 & x \bmod 2 = 0 \\ 1 & x \bmod 2 = 1 \end{cases}$$

- 当 p 为偶数时

$$sg(x) = \begin{cases} 0 & x < p \ \& \ x \bmod 2 = 0 \\ 1 & x < p \ \& \ x \bmod 2 = 1 \\ 2 & x = p \\ sg(x \bmod (p+1)) & x \geqslant (p+1) \end{cases}$$

C. Battle

题解

第二种方式是分析这个游戏的不变点

- 当 p 为奇数时，其非负整数次幂均为奇数，两次操作过后总石子的奇偶性不会发生变化。因此奇数石子对应的 sg 为 1，偶数石子对应的 sg 为 0
- 当 p 为偶数时，一种可行的解释是在 $\bmod (p+1)$ 的角度下分析操作。由于 $p^k \equiv (-1)^k \pmod{p+1}$，从对 $(p+1)$ 的余数来看，只有加一和减一两种本质不同的操作。

C. Battle

题解

当剩余石子大于等于 $(p+1)$ 时，后手总是可以保证两次操作过后剩余总石子对 $(p+1)$ 的余数不变，因此 $sg(x) = sg(x \bmod (p+1))$

而当剩余石子小于 p 时，双方每次都只能取 1 个石子，胜负由奇偶性决定，sg 函数显然，这也就对应着上述公式中的第一、二行。

当剩余石子等于 p 时，其既能转移到 sg 为 0 的 0，又能转移到 sg 为 1 的 $(p-1)$，其 sg 为 2。

D. stack-out

题意

求出栈序列方案数，要求至少存在一段连续出栈 k 次的区间。

D. stack-out

题解

$dp_{i,j}$ 表示前 i 个元素目前多出 j 个左括号时，最大连续出栈长度小于 k 的方案数。

转移的时候，先正常从前一个元素转移一步，然后去掉恰好长度等于 k 的方案数，即从前 $k+1$ 步的位置，先入栈一次然后连续出栈 k 次的方案数（稍微考虑一下 $i=k$ 的边界情况）。

最后拿总方案数（dp 或卡特兰数求出）减一下即可。

时间复杂度:$O(n^2)$

E. segment-tree

题意

给一个线段树的一个查询区间构建的树。两个人玩游戏，一个人只能拿左链，即每对相邻点必须一个点是另一个点的左孩子，类似的另一个人只能拿右链，要求每个人至少拿 2 个点，拿完后树不能分裂，1 号点只能最后拿。不能拿的输，问胜负。

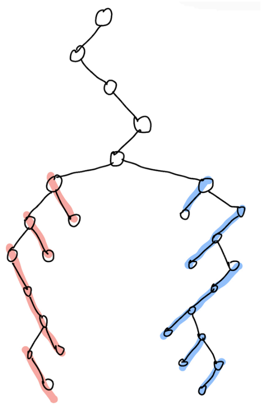
E. segment-tree

题解

线段树结构题，实际上每个人没有决策，模拟拿的过程即可。（注意区分一次拿两个还是拿三个的细节）

注意查询区间的树实际上是一个类三叉结构，它最多只会在一个节点处分裂，此后不再发生分裂。分裂后的两侧实际上类似于完整的前后缀查询。

以左侧分裂为例，如果一个点左孩子在区间树上，那么它的右孩子不会产生递归，并在左孩子上继续递归，否则从右孩子上继续递归。



F. Cities

题意

n 个城市在一个数轴上互相配对，每条边有上限，问所有方案的路径之和。

F. Cities

题解

令 $f_{i,j}$ 为前 i 个城市还有 j 个城市没有配对的方案数，$g_{i,j}$ 为前 i 个城市还有 j 个城市没有配对、已经配对的城市路径之和。每条边的限制等价于将 $j > s_i$ 的部分删去。

转移的话分两种情况，城市 $i+1$ 和前 i 个城市中的某一个待配对的城市配对，或者将城市 $i+1$ 加入待配对的城市中。

时间复杂度：$O(n^2)$

G. Copy and Paste

题意

有一个剪切板和一个输入框，有三种操作：

1. 在末尾输入一个字符
2. Ctrl-A Ctrl-C
3. 在末尾 Ctrl-V

问生成字符串 s 的最小操作数

G. Copy and Paste

题解

考虑 DP，设 dp_i 表示生成前缀 $S[1,n]$ 的最小步数，那么有：

$$dp_i = \min \begin{cases} dp[i-1]+1 \\ dp[j]+1+i-j-(ocr_{j,i}-1)*(j-1) \end{cases}$$

其中 $ocr_{j,i}$ 表示 $s_{1\ldots j}$ 在 $s_{1\ldots i}$ 中出现的次数（要求不交），发现 $ocr_{j,i} \leqslant \lfloor i/j \rfloor$，$\sum_j ocr_{j,n} = O(n\log n)$ ，i 从左向右扫描，这个数组只会进行 $n\log n$ 次单点 $+1$。

G. Copy and Paste

题解

维护 $last_j$ 表示 $s_{1\dots j}$ 在匹配到 $s_{1\dots i}$ 时最后一次匹配的位置，对于递增的 i ，现在要做的是：

对于所有的 $j \in \text{border}(i)$ ，如果 $last_j + j \leqslant i$ 则令 $ocr_j + 1$ ，$last_j = i$

这个可以使用树链剖分维护，对于每个 i 查询它在 border 树上到根的路径，如果发现 $last_j + j \leqslant i$ （这只会出现 $n \log n$ 次），则对这颗线段树进行修改，时间复杂度 $O(n \log^2 n)$

H. Permutation

题意

定义对排列的操作为：将一个排列前半部分和后半部分进行归并排序得到新的排列。

每次询问是否存在排列经过操作后可以得到指定排列。

H. Permutation

题解

我们考虑前缀 max 的位置 $l_1 < l_2 < \ldots < l_k$，如果在序列 A 选了 p_{l_i}，那么序列 B 开头的元素一定比 p_{l_i} 大，$[l_i, l_{i+1})$ 这一段一定在序列 A 中。

那么问题就转化成了一个多重集合 S，是否存在 S 的子集使得子集和为 $\frac{n}{2}$。

有一个很重要的性质，多重集合 S 所有数的和是 n，这意味着 S 中本质不同的数只有 $\sqrt{n}$ 个。我们对这 $\sqrt{n}$ 个数做多重背包即可。

时间复杂度：$O(n\sqrt{n})$

但有部分队伍用常数优秀的 bitset 以 $O(n^2/64)$ 通过了本题

I. Tree

题意

两种操作，将一棵树一条路径上的边权异或上一个数，或者询问一个点周围所有边权的异或和。

I. Tree

题解

签到题

我们对每个点直接维护答案，发现对于操作 1，路径上两个端点之外的点答案不变。因此操作 1 直接对两个端点异或上这个数，操作 2 直接输出即可。

时间复杂度：$O(n)$

J. Function

题意

给出一个正整数 $n(1 \leqslant n \leqslant 10^5)$ 和 n 个二次项系数为 1 的二次函数，第 i 个函数形如 $y=(x-i)^2+b_i(1 \leqslant b_i \leqslant n)$

然后给出一个正整数 $m(1 \leqslant m \leqslant 10^5)$，表示操作总数。

操作共有两种，第一种是添加一个二次项系数为 1 的二次函数，形如 $y=(x-a)^2+b(1 \leqslant a, b \leqslant n)$。第二种是询问所有二次函数在 $x=a(1 \leqslant a \leqslant n)$ 处的最小函数值。

具体的，每次操作会先给出操作的类型，如果是 0 表示是第一种操作，如果是 1 表示是第二种操作。对于第一种操作，会再给出两个正整数 a, b。对于第二种操作，则会再给出一个正整数 a。

J. Function

题解

查询 $x = a$ 时函数的最小值，容易发现可能为最小值的函数只可能在左右根号的范围内。

那么只要在左右根号的范围内枚举所有函数，找出最优的即可。

时间复杂度:$O(n\sqrt{n})$

K. Split

题意

给出一个正整数 n 和一个长度为 $n(3 \leqslant n \leqslant 10^6)$ 的单调不升序列 $a_i(\forall a_i \geqslant a_{i+1})$

然后给出一个正整数 $m(1 \leqslant m \leqslant 10^6)$，表示操作总数。

操作共有两种。第一种是将 a_x 变成 $a_{x-1} + a_{x+1} - a_i(1 < x < n)$。

第二种是询问操作。假设将序列分为 $k(1 \leqslant k \leqslant n)$ 段 (每段长度至少为 1)，每段的价值为最大的数减最小的数，求所有分段方案中最小的 k 段价值和。

K. Split

题解

先只考虑询问操作，我们容易发现，当在 i 和 $i+1$ 中间分割开，a_i 是前一段的最小值，a_{i+1} 是后一段的最大值，那么对答案的贡献就是 $a_{i+1}-a_i$。那么直接将差分数组排序，取前 $k-1$ 小即可。

然后考虑修改操作，容易发现，这修改就是交换差分数组相邻的两个元素，那么对答案就没有本质影响。

时间复杂度:$O(n\log n)$

L. Zhang Fei Threading Needles - Thick with Fine

题意

签到

题解

输出 $N - 1$