个人刷题

fks

CF1698G

题解：考虑抽象，简化过程。

1.首先前导0没用，只是最后移位。

2.第一个1肯定是要在的。而且整个过程是唯一的。（一旦得到第二个1，那就停止）

这个过程可以看成乘法过程。（异或看成在2域下的加法）。

于是S(x)*(sigma xi)=x^k+1

令后面的sigma xi=F(x)那么考虑只关心x^k那么直接对S(x)取模

于是就变成了x^k=S(x)-1(%S(x))

于是变成bsgs但是取模需要定义。

考虑取模实际上是T(x)*S(x)+G(x)=F(x)*x 我们要求G(x)

假设两个多项式deg相同，那么取T(x)=1, 于是G(x)=x*(F(x))-S(x)=x*(F(x))+S(x)

如果不一样，那么就 T(x)=0,即可。

CF1699E

他给出了值域的sigma作为限制，可以比较自然的联想到枚举最小值。考虑到我们可以通过固定最小值，求出最小的最大值，来最小化极差。我们可以先枚举最小值，然后用dp或者贪心来做所有数分解成大于这个最小值的因子，最小的最大值。

我的第一思路是觉得，贪心不行，选择dp

再考虑，一个数，在最小数变化的过程中，最大数会变化的地方当且仅当是出现在，最小值是因数的情况下。

那么我们相当于维护折线段可以做？

或者是，通过这个条件，我们发现，我们可以调和级数复杂度来做（因为在枚举最小值的过程中，我们只要枚举倍数，就能知道他影响了哪些数，或者是会造成哪些dp值的更新）

考虑dp的更新dp[x]表示x这个数，分解成大于等于lim的因子，最小的最大因子是多少。

dp[x]=min(dp[x],dp[x/lim],dp[x/lim/lim]...);

可以多重背包优化。相当于dp[x]=min(dp[x],dp[x/lim])

CF1700E

。。。人傻了。最基本的都没想到。

观察1：一个图满足条件，当且仅当每个点四周都有至少一个比他小的数。

那么我们显然可以把不满足条件的点取出来（选其中之一），把这个点周围的4个再加上他本身，这五个点中其中一个必然要交换，再另选一个点，模拟交换即可

CF1700F

待填坑

CF1704E

首先，考虑到如果所有a都是正的，或者是0的只有在“ 末尾 ”，那么满足一个性质，就是流光的顺序按照拓扑序。

我们考虑，如何计算这种优美的情况，一个点如果要流光，那么他会接受来自上游点的流量。并且我们不需要考虑上游点什么时候把东西全部给他，只需要知道上游会给他多少（因为他在流完前，上游的点一定会把东西全部给他，并且全部流完，也就是说我们让这个点每次以1的速度流失就好了，不需要考虑每个时候的量，只需要知道他要留流所有上游的流量的和）因为上游有多少，他就会原封不动的传给下游（虽然不是一次性，但总的效果是一下子给的）

CF1704F

比如对于Alice，首先考虑到，操作的时候R肯定会减少，要尽量让R减少的慢，但又要影响Bob，

我们可以有如下决策，除非全是R，那么不要选RR，在有RB的情况下，选RB，因为会让B变少），没有RB选RW，

所以，两个人对于RB的争夺就比较明显了。

先把所有RB取完，然后就独立开了，最终结果只和R和B的数量有关。而取RB的时候，R，B的差值不变，故如果一开始R和B数量不一样，那么多者胜。

如果一样。那么考虑，RB取完以后，先取者败。也就是把问题简化为取RB的过程。

而RB我们只考虑极长子段，所有子段是独立的，最终结果就是把所有的sg 异或起来即可。

我们把问题简化为研究一段长为L的RB子段的sg值。

我们考虑染其中一段，会把问题分裂成两个小的独立子问题。于是转移方程可以写出。

打表发现是循环的（本质是八进制博弈？？？）

个人学习

From:
https://wiki.cvbbacm.com/ - **CVBB ACM Team**

Permanent link:
https://wiki.cvbbacm.com/doku.php?id=2022-2023:teams:fire_and_blood:week_summary_2

Last update: **2022/12/02 17:18**