

个人刷题

fks

CF1687C

题解：考虑转化题意，令 $c[i] = b[i] - a[i]$ ，再对 c 作前缀和，那么就转化为了，每次对于 $[l, r]$ 如果 $c[l-1] = c[r]$ ，那么把 $l-r$ 这段区间全部覆盖为 $c[l-1]$ 。要求我们最后能把所有 c 都变成0。我们倒着考虑，来看操作是否有用（因为一开始我对于区间两两交的影响很头痛）。考虑最后反正要都变成0。那么必然 $c[l]$ 和 $c[r]$ 也必须要是0。否则做了和没做一个样。那么我们只对是0的考虑。我们把操作存在两个端点的vector里。暴力判断和更新就好了。用set维护非0位置弹出，set用于均摊，只染色非0位置）。

CF1687D

题解：一开始想的，把段都暴力弄出来，后面扫描线做。但发现其实不需要，考虑可爱的区间是 $[k^2, k^2 + k]$ ，那么我们就发现，当我们固定了 $a[1]$ 的段，也就是 $a[1]$ 的偏移量的范围确定，后面的段的偏移量也能唯一确定（因为一个段最多从可爱到不可爱，或者从不可爱到可爱，不可能跳变两次（凸性））。那么我们可以发现，一个在不可爱段的最小值，会对下界有影响。一个在可爱段的最大值，会对上界有影响。我们预处理出前驱后继，直接做。每次跳的次数是 V/i ，那么就是调和级数

CF1687E

题解：考虑给出的形式，实际上是每个因子的min和min_rk2相加，我们考虑他的选择方式，实际上暗示着我们考虑min_Max容斥，比较容易的可以得到式子（用广义minmax可以得到），然而我们发现复杂度是 $2^{n \times n}$ 无法通过。似乎没法优化？我们换个方向，想想能否减小n来简化问题。也就是说挑选出一些代表性的数，来与我们整个数列等价。我们考虑一个定理：一个数最多的因子个数是 $w(n)$ ，在 $1e6$ 内，这个函数是7。那也就是，我们可以每次钦定每个因子次小/最小给他选上。

CF1717E

题解：比较关键的一点就是想到枚举gcd（不枚举比较难做）。利用辗转相减，我们可以发现 $a=x*t, b=y*t, gcd(n-(x+y)*t, t)$ 就变为了 $gcd(n, t)$ 。但我们考虑要求出 $gcd(x, y) == 1$ 并且 $x+y=c$ 的对数，考虑 $y=c-x, gcd(x, c-x) == 1 \Rightarrow gcd(x, c)=1$ ，也就是欧拉函数

CF1717F

(vp) 题解： $+1, -1$ 比较难处理，考虑先全部 -1 ，做单选题，每次选一个 $+2$ ，那么就变成了考虑 $+2$ 分配给谁的问题。我们做个delta，把目标值和当前值作差，考虑每个人要被加几次 $+2$ ，也就是 $\text{delta}/2$ 。那么可以判断掉一些不合法的 delta 负、 delta 不能被2整除。比较显然可以得到网络流模型，但我们考虑到有些是无限制，如果按照“最多”的约束连边，则能达到最大流，但没法满足恰好。考虑上下界网络流建图即可。

ljz

VP的签到题就不写了。

CF1730D

题意：每次可以把a的前k位和b的后k位交换，问a是否可以变成b。 题解：容易发现，对于 a_i 和 b_{n-i+1} 这两个元素的对应关系不管怎么操作都是不会变的。所以有解无解的判断方法就是相同的pair（无序）有偶数个，或者有一个奇数且n是奇数。

CF1730E

题意：计算有多少个区间满足最大值可以整除最小值。 题解：先用单调栈求出以每个元素为最大值最小值的最长区间。我们对每个 a_i 都求出以 a_i 为最大值的满足条件的区间，枚举 a_i 的每个约数 d （ a_i 在一百万以内因此可以AlogA预处理），对于一个约数 d （可以找到离 a_i 最近的 d 左右各一个），我们只需要计算包含 a_i 和 d 的满足条件的区间，对于左右的 d 注意去重。

CF1730F

题意：给定一个排列 p 和 k 找到一个排列 q 满足任意 $i < j$ 都有 $p_{qi} \leq p_{qj} + k$ 使得 q 的逆序对数量最少 $(n \leq 5000, k \leq 8)$ 题解：我们一个一个考虑 q 对于 q_1 的值只可能是 $[1, k+1]$ 中的一个，如果 p_{q1} 取了1那 p_{q2} 的值只可能是 $[1, k+1]$ 中的一个，如果 p_{q1} 没有取1那么 p_{q2} 的值还是只能在 $[1, k+1]$ 中取，后面以此类推，也就是说每放置一个 q 我们需要考虑的值只有最小的没被选过的数到这个数+k的范围内，这样就可以状压DP了。

CF1733E

题意： $n*m$ 的方格图上每个格子有一个箭头，初始时 $(0,0)$ 上有一个小球，以后每一个时刻所有小球都会按照箭头的方向走一个格子，然后 $(0,0)$ 上会新出现一个小球，所有上一个时刻有球的格子都会转换箭头方向（只有向右和向下两种状态）。问第 t 时刻给定的格子是否有球 $(t \leq 1e18, x_0, y_0 \leq 120)$ 题解：我们把所有小球统一考虑，也就是说一共会生成 t 个小球，小球走到 (x_0, y_0) 需要 $x_0 + y_0 - 1$ 个时刻，我们考虑有可能经过 (x_0, y_0) 的小球一共 $t - x_0 - y_0 + 1$ 个，如果经过当前格子的小球有 n 个，那么其中会有 $\lfloor \frac{n}{2} \rfloor$ 个经过下方的格子，剩下的会经过右边的格子。我们只需要知道 t 时间经过 (x_0, y_0) 的小球数量和 $t-1$ 时间经过 (x_0, y_0) 的小球数量是否相等就可以判断是否在 t 时刻 (x_0, y_0) 上有球。

CF1720D

CF1717E

CF1717F

CF1716E**CF1716F****CF1715F****CF1715E****CF1713E****CF1712E****CF1709E****CF1706D****CF1697E****CF1696F****CF1696G****ARC149D****ARC149E****ARC149F****ABC271G****ARC148E****个人学习****ddp**

新学了ddp[]目前只做了P4719[]还未学全局平衡二叉树[]ddp实际上就是维护了一段重链的转移矩阵，难点在于矩阵的设计和dp状态的设计，可能有时候dp状态会为了转移，把一些权值给并入，比较抽象。需要维

Last
update:
2022/10/12 2022-2023:teams:fire_and_blood:week_summary_1 https://wiki.cvbbacm.com/doku.php?id=2022-2023:teams:fire_and_blood:week_summary_1&rev=1665542259
10:37

护轻儿子的dp值和当前的dp[]每次更新一条重链的权值，再把更新重链顶端的父亲的权值。由于每个点维护了轻儿子的dp[]且转移矩阵只和轻儿子的dp有关，所以每次修改的矩阵数只有log

updata(2022.10.12)有了点新的理解，全局平衡二叉树感觉就是[]lct用加权重心建树的方式，把每个平衡树建出来（权重是一个节点对应轻子树的大小）。然后操作就和lct一样[]access的时候不需要splay[]只需要上传信息即可

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_1&rev=1665542259

Last update: **2022/10/12 10:37**