随机

散列表

- 又称哈希表

- 1.挂链法
- 2.向后找第一个空位置

- 一张无向图，若一个点集，其内的点两两之间都有边直接相连，这个点集就是一个“团”。
- 求最大团。
- $n \leq 40$
- 4 4
- 1 2
- 1 3
- 2 3
- 1 4
- 3

- 随机化：
- 随机一个排列，按照排列的顺序尝试向团内加点，若当前点与当前团内所有点都有边则加入，否则不加入。随机多次。
- 状压DP：
- 设 f_S 为只保留点集 S 的最大团大小。
- 设 S 中的第一个元素为 x 。
- 1.不选 x： $f_{S \wedge (1 \ll x)}$。
- 2.选择 x： $f_{S \& g[x]} + 1$， $g[x]$ 为与 x 相连的点集状态表示。
- 复杂度?
- 去掉前一半点时至多 $\frac{n}{2}$ 步，每一步两种选择，至多 $2^{\frac{n}{2}}$ 。
- 后一半点状态数为 $2^{\frac{n}{2}}$ 。
- cf1105E

- 一个序列 A，要支持两种操作。
- 1.单点修改
- 2.给出 l_i, r_i, k_i 询问区间 $[l_i, r_i]$ 内所有数字的出现次数是否都为 k 的倍数。
- $n, m \le 3 \times 10^5$
- 6 3
- 4 2 3 3 2 1
- 2 1 6 2
- 1 1 1
- 2 1 6 2
- NO
YES

- 若答案为YES，则对于所有数字的任意子集，它们的总出现次数也是 k_i 的倍数。
- 随机一个集合，对于 $k_i = 2$ ，出错概率为 $\frac{1}{2}$ 。
- 先随机出 m 个数字集合，每个集合用BIT维护其区间出现次数，询问时若这 m 个集合的出现次数都是 k 的倍数，就输出YES。
- m 不少于30即可。

- cf1746F

- 给定 n 个数字 a_i 。
- 求最大的 g ，使得至少有一半的 a_i 是 g 的倍数。
- $n \leq 10^6, a_i \leq 10^{12}$
- 6
- 6 2 3 4 5 6
- 3

- 随机一个 a_i ，它成为 g 的倍数的概率至少为 $\frac{1}{2}$ 。
- 随机多次，暴力检查每个约数?
- 10^{12} 范围内约数个数最多的有上千个。
- 对于一个 a_i 和其某个约数 g ，若 a_j 也是 g 的倍数，则有 $\gcd\left(a_i, a_j\right)$ 是 g 的倍数。
- 统计不同的 $\gcd\left(a_i, a_j\right)$ 个数。
- $\gcd\left(a_i, a_j\right)$ 都是 a_i 的约数，直接枚举并统计即可。

- cf364D

- n 张卡牌，每张卡牌有三个值 p_i, c_i, l_i 。
- 你需要选出一个卡牌集合，满足：
- 1.选择的卡牌的 p_i 之和不小于一个给定的 k 。
- 2.若选择了卡牌 i, j ，则 $c_i + c_j$ 不能是质数。
- 最小化选择的卡牌中 l_i 的最大值。
- $n \leq 100$
- 3 7
- 4 4 1
- 5 8 2
- 5 3 3
- 2

- 答案具有可二分性。
- 随机化：
- 每次随机一个排列，能加入就加入。
- 确定性做法：
- 特判1，同样奇偶性的数相加后为不少于4的偶数。
- 冲突只会发生于奇数和偶数之间。
- 二分图最大独立集。

- cf808F

- 给定 n 个固定区间 $[l_i, r_i]$ 和一个 m，你可以选择 $1 \le L \le R \le m$，我们称 $[L, R]$ 是好的是指，它与至少一个固定区间相交，且对于每一个固定区间 $[l_i, r_i]$，若 $[L, R]$ 与 $[l_i, r_i]$ 相交，则相交的长度必须为奇数，若不相交则无限制。你需要求出所有好的 $[L, R]$ 的长度之和。
- $n, m \le 2 \times 10^5$
- 3 6
- 2 4
- 3 6
- 4 4
- 19
- (1,2),(1,5),(2,2),(2,5),(3,3),(4,4),(4,6),(5,5),(6,6)

- 对于每一个固定区间：[是否相交(0/1)]+相交长度为偶数。
- 对每一个固定区间 $[l_i, r_i]$ 赋一个随机权值 x_i，偶数个 x_i 的异或和为0。
- 相交长度个 x_i 的异或和？设数组 A， $A_k = A_k\ xor\ x_i (l_i \le k \le r_i)$，$A[L]\ xor\ A[L+1] \dots A[R]$ 即为所求。设 A 的前缀异或和为 B。
- 判断是否相交？ $l_i \le R\ and\ r_i \ge L$。
- 即所有 $l_i \le R$ 的固定区间去掉所有 $r_i < L$ 的固定区间。
- 设 $l_i \le t$ 的固定区间的 x_i 的异或和为 LV_t，同理有 RV_t。
- 则一个区间 $[L, R]$ 合法为 $B_R\ xor\ B_{L-1}\ xor\ LV_R\ xor\ RV_{l-1} = 0$，即 $B_R\ xor\ LV_R = B_{L-1}\ xor\ RV_{L-1}$。
- 从左往右枚举 R， LV, RV 可以顺路维护。
- 长度之和？维护数量，下标和。 $\sum (R - L_p) = R \times num - \sum L_p$。
- cf799F

- 给定 n 个数字 a_i ，对于每个数字，一次操作可以将它+1或是-1，同时要保证操作后的数字仍是正数。
- 问最少多少次操作，使得所有数字的最大公约数大于1。
- $n \leq 2 \times 10^5, a_i \leq 10^{12}$
- 5
- 9 8 7 3 1
- 4

- 9 9 6 3 3

- 考虑让所有数字都变为偶数，至多需要 n 次操作。
- 操作不少于两次的数的数量至多为 $\frac{n}{2}$ 个。
- 操作不多于一次的数的数量至少为 $\frac{n}{2}$ 个。
- 随机取一个 a_i ，操作后最终得到 $a_i - 1, a_i, a_i + 1$ 的概率至少为 $\frac{1}{2}$ 。
- 枚举这三者的所有质因子来检查。
- 多次随机取 a_i 。

- cf1305F

- 一个01串，取其两个连续子串（可以有重叠部分），使其作为二进制数按位或的结果最大。
- 01串是随机生成的。
- $n \leq 10^6$
- 1110010

- 1111110

- 1110010 OR 11100 = 1111110

- 1.肯定是取两个前缀。
- 2.其中一个一定是整个串。
- 另一个串要尽可能增加按位或之后的结果，因此最好把最靠前的几个0变成1。
- 最靠前的连续1的数量在随机情况下是 $O(1)$ 的。
- 要使最靠前的0变为1，这样的前缀数量也是 $O(1)$ 的。

- cf1743D

- 有 n 个在 $[0,1]$ 范围内随机的实数 x_i ，再给定 m 条限制条件，为如下两种之一：
- 1. $x_i + x_j \leq 1$ 。
- 2. $x_i + x_j \geq 1$ 。
- 求满足所有限制条件的概率。
- $n \leq 20$
- 3 2
- 0 1 2
- 1 3 3
- $x_1 + x_2 \leq 1, x_3 + x_3 \geq 1$ ，概率为 $\frac{1}{4}$ 。

- 将 $x_i < 0.5$ 的 i 染为白色， $x_i > 0.5$ 的 i 染为黑色。
- 对于一条限制，若出现在同色节点之间，则可以直接判断合法性。
- 若1号限制出现在白色节点 i 和黑色节点 j： $x_i + x_j \le 1$ ，也就是 $x_i \le 1 - x_j$ ，对于2号限制，就是 $x_i \ge 1 - x_j$ 。1号限制相当于 i 向 j 连一条有向边，2号限制相当于 j 向 i 连一条有向边。
- 设 $y_i = \min(x_i, 1 - x_i)$ ，要满足所有的限制条件，相当于 y_i 要按照该有向图的某一个拓扑序从小到大排列。
- 2^n 枚举染色方案，构图计算拓扑序数量?
- 枚举拓扑序，1号限制相当于强制拓扑序靠前的点为白色，2号限制相当于强制拓扑序靠前的点为黑色。
- 从后往前确定拓扑序的每一位，设已用状态为 S ，当前位置可能需要被染色（取决于已确定的靠后的点），或自由染色。
- $O(2^n n)$
- cf1842H

- 给定一个长为 n 的排列 a_i ，初始时你站在数轴的0处，并在该位置写下了 a_1 。接下来的第2到第 n 步，每次进行如下操作：
- 1.选择不动，或是向左移动一个单位，或是向右移动一个单位。
- 2.将数字 a_i 写在当前位置，若已有数则进行覆盖。
- 最终，将数轴上写下的数从左至右按顺序取出，你需要最大化取出的序列的最长上升子序列。
- 排列 a_i 随机生成， $n \leq 15000$ 。
- 4
- 4 1 2 3
- 3

- 倒序考虑，就不会出现覆盖的情况了，只有在当前位置为空时才会写入数字。
- 对于不在最终的最长上升子序列中的数，没有必要写到数轴上（除了 a_n ），因为更短的长度更利于前后移动。
- 设 $f_L(k,i)$ 表示第 i 个数被放在了当前序列的左端，当前总共写上了 k 个数字的情况下，序列右端数字的最小值。
- 设 $f_R(k,i)$ 表示第 i 个数被放在了当前序列的右端，当前总共写上了 k 个数字的情况下，序列左端数字的最大值。
- 考虑下一个要被写上的数字，若与 i 写在同一端，直接转移，否则若 $i-j \geq k$ 且 a_j 能放到另一端，交叉转移 f_L, f_R 。
- 线段树优化就可以做到 $O(knlogn)$ ，注意到排列随机生成，最长上升子序列长度为 $O(\sqrt{n})$ ，复杂度变为 $O(n^{1.5}logn)$ 。
- cf1530H

- 给定一个长为 n 的序列 a_i ，多次询问 $[l_i, r_i]$ ，最少将区间 $[l_i, r_i]$ 中的元素分为多少个集合，使得每一个集合中出现次数最多的数的出现次数不超过该集合大小的一半（上取整）。
- $n, m \leq 3 \times 10^5$
- 6 2
- 1 3 2 3 3 2
- 1 6
- 2 5
- 1
- 2

- 找到区间内出现次数最多的数字 x ，若其出现次数 t 不超过该区间长度 l 的一半，则答案为1，否则将剩余的 $l-t$ 个数和 $l-t+1$ 个 x 组成集合，剩余的 x 每个单独一个集合，集合数为 $2\times t-l$ 。
- 在区间内随机一个数，是 x 的概率至少 $\frac{1}{2}$ 。
- 随机若干次，若没有出现超过一半的，则认为不存在 x 。

- 若数字 x 在区间中的出现次数至少为区间长度的一半，则将区间分割为若干个子区间后， x 至少会在某个子区间中出现至少该子区间长度的一半次。
- 线段树，每个节点维护 x ，从左右儿子中取出现次数更多的那个。
- 不一定是众数，但一定能找到出现至少一半的数。
- $O(nlog^2n)$
- 抵消法，扫描序列，记录当前值和数量，相同值+1，不同值-1，若有数出现超过一半次，最终一定剩下。
- 线段树维护， $O(nlogn)$ 。

- cf1514D

- 给定一个长为 n 的序列 a_i ，若你当前在 i 号点上，你可以跳跃到 $[i, i+a_i]$ 中的任何一个点。
- m 次询问，最多删掉 k_i 个点的情况下，从 l_i 开始，至少跳几步可以跳到 r_i 。
- $n, m \leq 20000, k_i \leq 30$

- 4 1
- 1 2 1 3
- 1 4 1

- 2

- 若当前不是最后一步，从 i 号点出发一定会跳到 $j + a_j$ 最大的 j，其中 $i \le j \le i + a_i$ 。
- 若要删掉 k 个点，可以认为一定会跳到 $i + a_i + k$ 。
- 倍增， $f[i][j][k]$ 表示从 i 出发，跳 2^j 步，删 k 个点，会跳到哪里。
- $O(nk^2logn)$

- cf1523H