

2022 牛客暑期多校训练营9

E-Longest Increasing Subsequence

构造一个 $1, 2, \dots$ 的排列，使其恰好有 m 个不同的最长上升子序列 $1 \leq m \leq 10^9, 1 \leq n \leq 100$
 将 m 二进制拆分，设 $m = a_0 2^0 + a_1 2^1 + \dots + a_k 2^k$ 其中 $a_k = 1$ 则可以构造
 $2^1 4^3 \dots (2k)(2k-1)$ 达到 2^k 再通过中间按顺序插入大于 2^k 的数以构造 $2^i, i \in [0, k]$ 最后在通过插入并调整大于 2^k 的数来维持上升子序列的长度一样。

F

题目大意：给定 $n \times m$ 的矩形，其中的元素是 $n \times m$ 的排列。对于某个子矩形来说，它的贡献为矩形内所有元素的 $\gcd$ 求所有子矩形的贡献和

如果要求贡献和，就只需要知道 $\gcd = i$ 的子矩形有多少个

可以先求 $\gcd$ 为 i 的倍数的矩阵数量，然后容斥

把所有为 i 的倍数的格子视为 1，其他视为 0，问题转化为在 0/1 矩阵中求全 1 矩阵数，这个可以用单调栈实现

G-Magic Spells

给定 n 个字符串 $s_1, s_2, \dots, s_n$ 求有多少个本质不同的公共回文子串 $(1 \leq k \leq 5, 1 \leq \sum |s_i| \leq 3 \times 10^3)$

对于第 i 个串在回文自动机上标记 2^{i-1} 每次插入新串时将 last 置 1，最后遍历所有状态，记录标记为 2^k 的状态个数作为答案。

K

题目大意：一共有 K 位，给定 n 个集合，定义一个集合的度数为 “从给定的集合中至少选出多少个集合才能得到该集合的超集”，求度数依次从 0 到 K 的集合数量

设 $F_i(S)$ 表示，用 i 个集合是否能选出 S 那么有 $F_i(S) = F_{i-1}(S) * F_1(S)$ 其中 $*$ 为或卷积 $F_1(S)$ 用 n 个给定集合初始化

设 $G(S)$ 表示得到集合 S 所需的最小给定集合数，可以在求 $F_K(S)$ 的过程中得到

把 $G(S)$ 的贡献下放到所有子集中，统计答案即可

Replay

A题是个签到 B题是一个差分+dp I题是一个dp 感觉都算是很水，没有太多好说的。

G题是个回文自动机，于是坐大牢了。

以后再不能写hash自然溢出了。

Dirt

无

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

Permanent link:
<https://wiki.cvbbacm.com/doku.php?id=2022-2023:teams:kunkunkun:2022-nowcoder-9> 

Last update: **2022/08/31 16:54**