牛客暑期 ACM 多校训练营

第1场 - ftiasch



• Monotonic Matrix

考虑 01 和 12 的分界线

是 (n, 0) 到 (0, m) 的两条不相交 (可重合)路径

平移其中一条变成 (n-1, -1) 到 (-1, m-1)

变成起点 (n, 0) 和 (n-1, -1), 终点 (0, m) 和 (-1, m-1) 的严格不相交路径

套 Lindström-Gessel-Viennot lemma

答案是 C_{n+m, n}² - C_{n+m, m-1} C_{n+m, n-1}

ISymmetric Matrix

把矩阵看成无向图的邻接矩阵,即要求所有点度为 2 设 f(n) 表示 n 个点满足条件的图的数量 $f(n) = (n-1) f(n-2) + \operatorname{sum}_{k < n-2} (n-1)! f(k) / k! / 2$ 设 $g(n) = \operatorname{sum}_{k < n-2} (n-1)! f(k) / k! / 2$ 则 g(n) = (n-1) g(n-1) + (n-1) (n-2) f(n-3) / 2

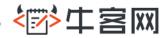
IFluorescent 2

设 X_i 表示灯 i 是否为亮 题目要求 $E[(sum X_i)^3] = sum_{i, j, k} Pr[X_i X_j X_k]$ 仅考虑 i, j, k 三列形成的矩阵 A则 $Pr[X_i X_j X_k] = 1 / 2^{rank(A)}$ 因为行秩 = 列秩,下面考虑列秩,同时忽略边界情况 枚举列 i,用 i 消其他列,复杂度 $O(nm^2)$ 此时,列j,k有2种情况: 列j和列k完全相同,则秩为2列j和列k不完全相同,则秩为3因此,只需把消元后的列归类即可(hash or trie) 总复杂度 *O*(*nm*²)



■Two Graphs

直接 n! 枚举可能的同构方案 Hash 去重即可 也可除以自同构的方案数



IRemoval

设 next(i, c) 表示位置 i 后第一个字符 c 的位置 f(i, j) 表示当前匹配到 i ,删了 j 个,不同的方案数 转移时枚举下一个字符 c ,转移到 f(next(i, c), j + next(i, c) - i)

ISum of Maximum

```
转成计算 \operatorname{sum}_k \operatorname{sum}_\mathbf{x} [\max\{\mathbf{x}\} > k] = \operatorname{sum}_k (a_1 \times a_2 \times ... \times a_n - \operatorname{sum}_\mathbf{x} [\max\{\mathbf{x}\} \le k]) = \operatorname{sum}_k (a_1 \times a_2 \times ... \times a_n - \operatorname{sum}_\mathbf{x} [x_1 \le k] [x_2 \le k] ... [x_n \le k]) 而 [\mathbf{x} \le k] 是个分两段的线性函数 所以后一项是个 n 段的关于 k 的 n 次多项式 每段拉格朗日插值即可
```



ISteiner tree

求最小边数的 Dreyfus-Wagner 算法

设 f(S, v) 表示集合 S 中的点到点 v 的最小边数

转移有2种:

固定 S, v 从相邻点转移,相当于根节点生长,复杂度 $O(2^k m)$ 固定 v, S 从 T + (S\ T) 转移,相当于节点分叉,复杂度 $O(3^k n)$ 为了避免重复计数,约定如下:

只计算以节点 1 为根的有根树 T 和 $S \setminus T$ 合并时,保证 T 的最小值小于 $(S \setminus T)$ 的最小值



Longest Path

先算 down[u] 表示 u 向下的最长路,接着算 up[u] 子问题是要求形如 opt[i] = max $_{j!=i}$ down[j] + (c[i] - c[j]) 2 可以按照 c[i] 排序,从前往后从后往前各做一遍 经典的斜率优化,也可用李超树



■Substrings

枚举 3!=6 种同构, 把字符串复制 6 份拼接得到 S

用 SA/SAM/ST 求 S 的不同子串

可以发现:

只有1种字符的串被算了3次

其他串被算了6次

答案是 (不同子串数量 + 3 × 单一字符的串) / 6



■Different Integers

离线按照 r 从小到大处理询问,考虑*没出现*的数字个数假设 r = last[x],那么当 l < first[x] 时,x 没出现用树状数组维护即可



Thanks

