The 2022 ICPC Shangdong Province Contest
Contest Editorial

Prepared by xxx

May 22, 2022

A. Seventeen

Task

使用 $1, 2, \cdots, n$, $+ - *$ 三种符号和括号构造表达式，使其值为 17。

A. Seventeen

- 当 $n \leq 3$ 时无解。
- 当 $n \leq 4$ 时有解：(5*3+4-2*1)。
- 当 $n \leq 5$ 时有解：(3*(2+4)-1)。
- 当 $n > 5$ 时，将 n 与 $n - 1$ 配对为 $(n - (n - 1))$ 的形式，乘在前面的表达式上，递归进行。

B. Minimum Expression

Task

给定一个长度为 n 的数字串（不含 0），在其中插入 m 个加号得到一个表达式，最小化这个表达式的值。

B. Minimum Expression

- 容易发现需要尽量平均地切分序列。令 $dp_{i,j}$ 表示切分的最后一个端点为 i，且已经切了 j 段时的答案。转移时枚举长度只有 $O(1)$ 种，复杂度为 $O(nm \cdot w(m \cdot 10^{n/m}))$，其中 $w(t)$ 为值域 $\leq t$ 的高精度的复杂度。
- 可以使用 python 实现。

Task

给定两个凸包 A, B，保证 B 严格被 A 包含。求在 A 中且与 B 无交的最大凸包面积。

C. Convex

- 枚举一条直线围绕凸包 B 旋转，将 A 的部分切割下来，切割部分的最大值即为答案。
- 在切割过程中按照直线与外凸包 A 的交的位置以及与内凸包 B 相切的顶点，可以分成至多 $O(n^3)$ 段，每一段内可以用简单的函数描述凸包的坐标变化以及其面积。

D. The Matrix

Task

给定 k 个带权简单连通图 G_i，记 $G = G_1 \times G_2 \times \cdots \times G_k$，两个点 $(v_1, v_2, \cdots, v_k)$ 和 $(v'_1, v'_2, \cdots, v'_k)$ 之间的有边当且仅当存在一个 j 使得除了 $\forall k \neq j$，$v_k = v'_k$，且 v_j, v'_j 之间有边。求这 G 的最小生成树 $\mod 998244353$。

D. The Matrix

- 模拟 Kruskal 的过程，将所有图的边取出排序，对于每一个图 G_i 维护并查集以及对应的连通块数 c_i。
- 对于当前边 u_i, v_i，若在其对应图中已经处于同一连通块则忽略。
- 否则连接这条边会使得 G_i 减少一个连通块，任意时刻将大图 G 看成 G_i 所有连通块缩点后的笛卡尔积，容易发现这条边会被计算 $\prod_{i\neq j} c_j$ 次。

E. Subsegments

Task

给定一个序列 $a_1, a_2, \cdots, a_n$ 和数 x，求有多少区间的乘积 $\bmod 998244353 = x$。

E. Subsegments

- 当 x 为 0 时，计算包含 0 的区间数即可。
- 当 x 不为 0 时，在每一个 0 处将序列断开，每一段内处理前缀积 s_i，即求 $s_i \times inv(s_j) = x$ 的个数，用 map 维护桶即可。

F. selection

Task

n 个人，比赛有若干轮，每轮对于区间 l, r 比赛，能力值最大的人获胜留下来，其他人退出。如果有相同能力的，随机选择一个人胜出（可以看做无人胜出，但是仍有一个假人回到序列中）。
给定 $n-1$ 个人的能力值，剩下一个人能力值未知，且他可以随意挑选初始站的位置。
维护 m 个操作：
1. 增加一轮 l, r。
2. 假设剩下一个人的能力值为 x，求他最多能赢几轮。

F. selection

- 每一轮看成是将 l, r 内的所有人缩点，可以得到 $[l, r]$ 对应初始序列的区间 L, R。仅当剩下的一个人初始在 L, R 内时有贡献，且能赢这一轮即他的能力值大于剩下 $n-1$ 个人中 $[L, R-1]$ 这些人的能力值。
- 离线，按照能力值排序，用线段树维护即可。

G. Corona Virus

Task

给定一棵 n 个点的树和阈值 k，求将树剖成若干连通块且连通块直径不超过 k 的方案数。

G. Corona Virus

- 令 $dp_{i,j}$ 表示 i 的子树内，连接到 i 的最长链长度为 j 的方案数。
- 合并 u, v 时，考虑断开 (u, v)，此时 $dp'_{u,i} \leftarrow dp_{u,i} \cdot \sum dp_{v,j}$。
- 保留 (u, v) 时，则 $dp'_{u,\max\{i,j+1\}} \leftarrow \sum_{i+j<k} dp_{u,i} \cdot dp_{v,j}$。
- 该转移式复杂度不高于上界为 k 的树形背包，即为 $O(nk)$。

H. Counting

Task

在一张 $n \times m$ 的网格图上，给定 k 个人在 $0 \sim T$ 秒内的行动路线（每秒只能走一步）。
求每秒重叠的人的对数。

- 对于网格图的每一个位置维护一个桶，答案即 $\sum \binom{c_{i,j}}{2}$。
- 根据每一个人的移动情况修改桶，同时维护答案。

I. gcds

Task

对于序列 a_i，定义权值 $f(a) = \sum_{i>1} \gcd(a_i, a_{i-1}) \cdot w(a_i)$，其中 $w(x) = px^3 + qx^2 + rx$。
求长度为 m，总和为 n 的序列的权值总和。

I. gcds

- 考虑每一个 a_i 和 a_{i-1} 对于答案的贡献，对于剩下的数通过插板法计算方案，即：

$$(m-1)\sum_i \sum_j \gcd(i,j) w(i) \binom{n-i-j-1}{m-3}$$

做欧拉反演/莫比乌斯反演可以得到答案为：

$$ans = (m-1)\varphi(d) \sum_{d|i} \sum_{d|j} w(i) \binom{n-i-j-1}{m-3}$$

容易通过欧拉筛计算 $\varphi(d)$，设 $f(d) = \sum_{d|i} \sum_{d|j} w(i) \binom{n-i-j-1}{m-3}$

$$f(d) = \sum_{d|i} \sum_{d|j} w(i) \binom{n-i-j-1}{m-3}$$

$$= \sum_{d|i} \binom{n-i-1}{m-3} \sum_{d|j, j<i} w(j)$$

- 对于 $w(x)$ 的每一项考虑，即：

$$f(d) = \sum_{d|i} \binom{n-i-1}{m-3} \sum_{d|j, j<i} (pj^3 + qj^2 + rj)$$
$$= \sum_{i} \binom{n-id-1}{m-3} \sum_{j=1}^{i-1} (pj^3d^3 + qj^2d^2 + rjd)$$

- 记 $s_k(n) = \sum_{i=0}^{n-1} i^k$，$s_k(n)$ 是一个 $k+1$ 次多项式，则：

$$f(d) = \sum_{i} \binom{n-id-1}{m-3} (d^3ps_3(i) + d^2qs_2(i) + drs_1(i))$$

计算 $f(d)$ 的复杂度为 $O(nk^2 + n\ln n \cdot k)$。

I. gcds

- 在上式中 $d^3ps_3(i)+d^2qs_2(i)+drs_1(i)$ 是关于 d,i 的表达式。
- 重整后，可以转化为关于 d,id 的表达式 $w(d,id)$（该多项式中 d 的指数可能为负），重写上式得到：

$$
\begin{aligned}
f(d) &= \sum_{i}\binom{n-id-1}{m-3}w(d,id)\\
&= \sum_{d\mid i}\binom{n-i-1}{m-3}w(d,i)\\
&= \sum_{t} d^t\sum_{d\mid i}\binom{n-i-1}{m-3}[d^t]w(d,i)
\end{aligned}
$$

则对于每一个 t，记 $g(d,t)=\sum_{d\mid i}\binom{n-i-1}{m-3}[d^t]w(d,i)$，可以看作是 Dirichlet 后缀和的形式，计算所有 $g(d,t)$ 的复杂度为 $O(nk^2+n\log\log n\cdot k)$。

J. Football Match

Task

给出了一面棋子的比例图和两个顶点的位置，求出旗子及内部五角星所有顶点的位置。

J. Football Match

- 几何题。可以先求出矩形和正五角星的中心，其它点可以依次推得：
- 正五角星的 5 个外端点是正五边形的端点，得到一个点后，其他点可以通过旋转角度求出；
- 内部的端点可以用正五边形端点两两连线后求交点的方法得到。

K. Coins

Task

Alice 有 4 种硬币 2,3,17,19，Bob 有 4 中硬币 5,7,11,13。
若干组询问 x，问谁能用更少的硬币凑出 x。

K. Coins

- 如果同时有 $a, b(1 < a < b)$ 两种面值的硬币，那么每 a 个 b 面值硬币可以用 b 个 a 面值货币替换，故 a 面值货币在最优解中 $< b$ 个。小范围内枚举或 dp 预处理，多余的部分用最大的硬币面值即可。

L. Segment

Task

给定序列 a_i，q 次查询，每次给定一组区间，求跨过仅保留这些区间时的顺序对数。

L. Segment

- 设定分块决策的阈值 C, 如果 $k > C$，则只有不超过 $\frac{m}{C}$ 个这样的询问，每个询问可以在 $O(n \log n)$ 时间统计；
- 对值域按块大小 B 分块处理 $k \leq C$ 的情况：
- 对每次查询，从左到右依次处理每个区间，查询值域每块内的值的个数，可以统计值域块间的贡献；
- 值域块内的贡献有 $O(nB)$ 个点对，扫描线扫序列，维护序列前缀中的点作为点对的第二个点，序列区间中的点作为点对的第一个点的方案数，可以枚举每次查询的两两区间（共 $O(mC)$ 个）计算贡献，右边的区间差分为两次前缀查询，查询左边的区间中的点作为点对的第一个点的方案数。
- 取 $C = \sqrt{n}$，$B = \sqrt{\frac{n+m}{\log n}}$，则总时间复杂度 $O(m\sqrt{n}\log n + n\sqrt{(n+m)\log n})$。

M. Travel Round the Grid

Task

在 $n \times m$ 的网格图上构造一个长度为 k 的环。

M. Travel Round the Grid

- none.

Thank you!