

「比赛总结」清北学堂省选十连测

Jiayi Su (ShuYuMo)

2021-03-29 20:54:14

之后应该会按照算法分类，后面有一些题目没有补完...

Round 1

Problem A (string)

你有一个字符串 S_0 ，想要用它造出共 n 个新字符串 S_0, \dots, S_{n-1} 。

你会按顺序进行 $n-1$ 次操作，造出新字符串。第 i 次操作有两种方法：

- **S x l r** 截取 S_x 的 $[l, r)$ 子串，即第 l 到第 $r-1$ 个字符作为 S_i
- **A x y** 将 S_x 和 S_y 拼接作为 S_i

最后，你会关心 S_{n-1} 长什么样，因此请输出 S_{n-1} 所有字符的 ASCII 码之和，结果对 $10^9 + 7$ 取模。

本题中字符串下标均从 0 开始编号，且保证所有字符串串长不超过 $2^{63} - 1$ ，并且所有操作均合法（即 $0 \leq x, y < i, 0 \leq l < r \leq \text{Len}(S_x)$ ）

第一行一个整数 n 。

接下来一行一个小写字母字符串 S_0 。

接下来 $n-1$ 行描述 $n-1$ 次操作，每行的操作格式见题目描述。

对于全部数据， $n \leq 2000, 1 \leq \text{Len}(S_0) \leq 2000$ ，同时满足题目描述中给的限制。

考虑如果存储每个串对应原串 S_0 的一段连续区间，每一次截取操作最多会产生两段不同的 S_0 区间，不妨利用这个思路，只要能够实现对操作的解析，复杂度不是问题，我的实现比较复杂...考虑把问题转化成一些简单的问题， $f(n, L)$ 表示查询第 n 个串的 $[1, L]$ 子区间的 Ascii 和。递归处理即可

应该时刻记住尝试使用递归这种方法。

Problem B (different)

你有 n 株植物，高度为 a_i 。你希望让它们的高度看起来不太相同，具体来说：任意两株高度差的绝对值不小于 d 。

每次你可以选择一株植物，将其拔高或压低 1，这花费你 1 的力气，这个操作可以进行任意多次。注意，植物高度不能是负的，即你需要保证任意时刻 $a_i \geq 0$ 。

你最少需要使用多少力气，才能使得植物高度看起来不太相同呢？

多组测试数据。第一行一个整数 T 为数据组数，接下来每组格式如下：

$\sum n \leq 3 \cdot 10^5, 1 \leq d \leq 10^6, 0 \leq a_i \leq 10^{11}$ 。

首先考虑一个转化，任意这样的限制不太好处理。考虑对原序列排序，然后第 i 株植物的高度减去 $(i-1) \times d$ ，然后问题转化为 对转化后的序列求最少需要支付多少代价，使得原序列不降。

考虑一个显然的 dp： $f[i][j]$ 表示考虑了前 i 个植物，第 i 个植物的高度为 j 的最小代价。

转移显然：

$$f[i][j] = \min_{k=0}^j \{f[i-1][k]\}$$

转移比较简单，设函数 $F_i(n) = f[i][n]$ 。

考虑每一个函数形态：

$F_1(n)$ 是一个绝对值函数，转移过程取前缀最小值中实际上是对斜率为正的后半部分“压平”，向下一个 $F_i(n)$ 转移其实是在原来的函数上加上去另一个绝对值函数，和一个常数。

这样形成的函数在任何时刻都是一个下凸壳，考虑使用堆来维护每一个拐点，如果在每次压入拐点的时候压入两个一样的拐点，就能够使得堆中的每个拐点前后斜率变化恰好为 1，每次取前缀最小值就相当于在堆中弹出函数值最大的拐点。

这样就可以实现堆维护转移了，再同时记一下 $F(0)$ 的函数值变化，最后遍历一遍堆种的拐点就能够取出最小值了。

需要分析 dp 特殊性质，然后从特殊性质入手进行优化。

Round 2

Problem A (grid)

你有一个 n 行 m 列的教室，其中某些位置不能坐人。

你的班上有 k 对 CP，你对八卦早已失去了兴趣，转而想要知道有多少种给这 $2k$ 位同学排座位的方式，使得每对 CP 的座位不相邻（有些位置可能会空出来，方案不同当且仅当某个位置状态不同，即坐着不同的同学）。

结果对 $10^9 + 7$ 取模。

对于全部数据， $1 \leq nm \leq 144, 1 \leq k \leq \frac{nm}{2}$ ，保证空位置个数不少于 $2k$ 。

考虑问题转化，不相邻不太好搞，但是如果钦定 k 个相邻，就可以直接 状压 dp 解决。

考虑一种摆放的方案，如果从上到下依次摆放这些 ** 每一次摆放的方案只和这个这个点和这个点周围的情况有关。考虑状压一个上轮廓即可。

设 $f[x][y][S][k]$ 表示考虑到了 (x, y) 这个位置，这个位置上轮廓的格子状态 S ，有 k 个配对的 **，的方案数。（这里可以先把 ** 考虑成无标号的最后统计的时候再给标号）。

转移只需要枚举是不放 / 竖着放一对 / 横着放一对。

将 $f[n][m][*][l]$ 累加后就能够得到无标号的 ** 配对 l 对的方案数。赋予标号，就能够得到钦定 l 的方案数。

可以用二项式定理转化为恰好 0 对匹配的方案数。

Problem B (cactus)

你有一棵仙人掌，你想要知道所有点两两最短路之和！即， $\sum_{i < j} \text{dis}(i, j)$ 。

仙人掌是一张无向简单连通图，无重边自环，且每条边属于至多一个简单环。

本题中边没有权值，即可以认为边权均为 1。

对于全部数据， $1 \leq n \leq 3 \cdot 10^5, 1 \leq m \leq 6 \cdot 10^5$ 。

屑题。

设仙人掌的点数为 n ，边数为 m ，则： $m \leq 2n$ 。

证明考虑一张图的 dfs 树，每一条不在树上的边都一定是返祖边（设这条边为 (u, v) ，则：要么 u 是 v 的祖先，要么 v 是 u 的祖先）。

对于一张仙人掌来说，每个返祖边无交，所以证明显然。

考虑每条边的贡献：

- 树边显然是断开后两连通块大小乘积。
 - 环上的边需要考虑每一条路径是经过这条边还是环上的对边。算贡献的时候不太好处理，可以考虑暴力算出一条边的答案，然后考虑环上下一条边和这条边的贡献会差哪些，这里就能实现 $\mathcal{O}(1)$ 转化为下一条边的贡献，屡试不爽。
-

Problem C (color)

你有一个 2 行 n 列的网格，你要给每个格子染上三种颜色（红绿蓝）之一。

你还提了很多要求：任意两个相邻格子颜色不能相同；任何一个 2×2 的块里，每种颜色都必须出现过。

现在你还想求出，恰好分别染了 R, G, B 个红、绿、蓝格子的方案数，结果对 $10^9 + 7$ 取模。

一行四个整数 n, R, G, B 。

对于全部数据， $2 \leq n \leq 5 \cdot 10^6, 0 \leq R, G, B \leq 2n, R + G + B = 2n$ 。

对于每一种染色方案，如果将每一列没有出现的字符写成一列，那么问题不难转化为求 限制三种字符的出现次数，然后排成一排使其满足相邻的两个字符不相同的合法方案数。

记录一种更加一般化的做法吧，直接考虑这样的问题：

给出 m 种颜色和其各自的出现次数 (a_i) 要求排成长度为 $n = \sum a_i$ 的序列，且满足相邻两个颜色相同的 恰好 有 k 对。定义两个序列不同，当且仅当存在某一个位置，两个序列颜色不同。

条件反射：看到 恰好 应该想到容斥，考虑如何求出钦定有 k 个颜色相同的相邻元素对。

需要观察到一个性质，对于某一种有 a_i 个的颜色，如果其在原序列中被分成了不连续的 x 段，那么她对答案的贡献是 $a_i - x$ 。

钦定 k 对 就可以转化为各个颜色一共被拆分成了 $n - k$ 段。这里的方案数就可以卷积统计了。

但是有一点点不太对，在有些情况下，颜色段的顺序交换后并不能算成不同情况，可以考虑给每种颜色相同的元素标号，最后对答案除以 $\prod_i a_i$ 。

考虑一种颜色 i 的生成函数，设： $[x^n]F_i(x)$ 表示第 i 种颜色拆成 n 段的方案数。

然后把这些段拼起来： $G(x) = \prod_i F_i(x)$ 。

组合起来之后，注意到这样是每一段是强制了其拼起来的顺序，需要打乱这个顺序，则钦定选 k 个的方案数就是 $(n-k)! [x^k]G(x)$ 。

然后套一个二项式反演就行了。

Round 3

Problem B (streeing)

给定一棵 n 个节点的树，树上的每个节点有一个字符 c_i 。

初始给出串 S ，现在有 Q 次询问 (x, y) ，你需要将 x 到 y 路径上的字符拼成一个字符串并回答串 S 在其中出现了多少次。

屑题。树上莫队 模板题

考虑每次新加入 / 减少一个字符最多会产生 / 减少一个匹配。哈希判断即可。

Problem C (Tree Queries)

给定一棵 n 个节点树，树上的每个节点有一个权值 a_i 。

现在有 Q 次询问 (u, d, k) ，你需要将所有距离点 u 小于等于 d 的点的点权排序并输出第 k 小的点权，如果没有第 k 小输出 -1 。

$$a_i \leq 10^9 \quad n, Q \leq 5 \times 10^4$$

屑题。毫无思维难度的一道省选难度的题。

点分树 + 主席树二维数点 模板题。

std 写的是两个 \log 的做法，就是考虑所谓的 同步二分。三个 \log 卡常数卡了一年拿到了暴力分。

Round 4

Problem A (matrix)

给定一个 $n \times m$ 的矩阵，矩阵的每个位置上都有一个整数。

你现在需要选择一个大小为 $1 \times x (1 \leq x \leq k)$ 或者一个大小为 $x \times 1 (1 \leq x \leq k)$ 的子矩阵并将其中的所有元素清零。

你想要使操作之后的最大子矩阵和尽可能大，输出这个值。

一个矩阵的子矩阵定义为连续的若干行与连续的若干列的交点对应位置构成的矩阵，子矩形可以为空。

对于所有数据， $2 \leq n, m, k \leq 300, -5000 \leq A_{i,j} \leq 5000$

比较套路的一个做法，考虑枚举上下两个边界，然后中间压成一维，预处理出每一列能够取到的最小能扣去的一列。设 $\text{sum}(n)$ 为前缀和函数， $d(l, r)$ 表示在 $[l, r]$ 这些列中最小的能够扣去的一竖条。

考虑 区间 $[L, R]$ 的答案一定是：

$$\text{sum}(R) - \text{sum}(L-1) - d(L, R)$$

可以预处理出 $d(n, n)$ 然后单调栈处理每一个区间的 d 值，然后使用 sum 初始化 ST 表，对于每一个 d 自然是取后半区间中最大的 sum 值，和前半区间中最小的 sum 值。能够做到复杂度 $\mathcal{O}(n^2)$ 。有亿点点小问题，细节没处理好，应该不是做法假了吧...

Problem B (Segment)

平面上有 n 条平行于 X 轴或平行于 Y 轴的线段，你要求出这些线段将平面划分成了几个部分。

第一行一个正整数 n

接下来 n 行每行四个整数 x_1, y_1, x_2, y_2 表示线段的两个端点。 $(x_1 = x_2 \text{ 或 } y_1 = y_2)$

保证 $(x_1, y_1) \neq (x_2, y_2)$

$1 \leq n \leq 10^5, -10^9 \leq x_1, x_2, y_1, y_2 \leq 10^9$

平面图的欧拉定理

$$V + F = E + 2$$

Problem C (3-sat)

请注意：本题是提交答案题

你有 n 个 01 变量 x_1, \dots, x_n 和 m 个子句 $C_i = (z_1 | z_2 | z_3)$ 或者 $C_i = (z_1 | z_2)$ 其中 z_1, z_2, z_3 为某个 x_j 或者 $\overline{x_j}$ (x_j)。每个子句的权值为 w_i 。（“|”为逻辑或）

你希望构造一组 x_1, \dots, x_n 的取值，使 $\sum C_i \cdot w_i$ 尽可能大。

第一行两个正整数 n, m

接下来 m 行每行 第一个整数 $k \in \{2, 3\}$ 描述这个子句的变量个数，接下来 k 个整数 z_1, \dots, z_k ，如果 $z_j < 0$ 则表示 $\overline{x_{-z_j}}$ ，否则表示 x_{z_j} ，描述一个子句，最后一个整数 w_i 表示这个子句的权值。

保证子句中的 $|z_j|$ 两两不同

最后一行十个正整数 $t_1 \leq t_2 \leq \dots \leq t_{10}$ ，描述评分参数。

设你求出的解 $\sum C_i \cdot w_i = \text{ans}$ ，令 v 为最大的 i 满足 $\text{ans} \geq t_i$ ，该测试点你将获得 v 分。

一行 n 个整数，表示 x_1, \dots, x_n 的取值。

认识了一下模拟退火算法，除了做提交答案题目以外，有一些不太紧的传统题也可以使用。

需要注意模拟退火算法，一定要设计出合理的邻域。需要注意，退火可能算是乱搞，但是代码不能乱写...，迭代次数是退火答案优劣的保证，需要加快邻域答案的计算，以追求更多的迭代次数。

Round 5

Problem A (星星)

三维空间里有一些星星，第 i 颗星星的坐标为 (x_i, y_i, z_i) ，且有 p_i 的概率发光， $1 - p_i$ 的概率不发光。保证任意四颗星星不在同一个平面上。

称一个星星是美丽的，当且仅当它是发光的，且不存在另外四颗发光的星星，使得这颗星星被严格包含在这另外四颗星星形成的三棱锥中。你要求出期望有多少星星是美丽的。

为了避免精度问题，你需要输出这个期望对 998244353 取模的结果。

第一行一个正整数 n ，表示星星个数。

接下来 n 行每行四个非负整数， p'_i, x_i, y_i, z_i 。其中 $0 \leq p'_i \leq 10000$ ，真实的 $p_i = p'_i \times 10^{-4}$ 。

对于所有数据， $4 \leq n \leq 100, 0 \leq x_i, y_i, z_i \leq 10^5$ ，保证不会有四点共面的情况。

Problem B (数字)

给定 n, p 以及长度为 $p-1$ 的数列 a_0, a_1, \dots, a_{p-2} ，其中 p 是个质数。你需要求出：

$$\sum_{i=1}^n a_{i \bmod (p-1)} \cdot \varphi(i) \bmod p$$

其中 φ 是欧拉函数，即 $\varphi(i)$ 表示小于等于 i 且与 i 互质的正整数个数， p 给定，且 p 是个小质数，满足 $2 \leq p \leq 23$ 。

总共会给定 T 组数列 a_i ，你需要对于它们分别算出答案。

对于所有数据， $1 \leq n \leq 10^9, 1 \leq T \leq 10^4, 0 \leq a_i < p, 2 \leq p \leq 23$ 且 p 是质数。

单位根反演

Problem C (糖果)

有 n 箱糖果，其中第 i 箱里有 a_i 堆糖果，每堆糖果都有 $2i$ 个，如果拿了 k 个，得到的快乐值为 $\sum_{j=\max(k,i)}^{2i} (-1)^{j-k} \binom{j}{k} \binom{i}{j-i} - [k=0]$ 。其中 $[k=0]$ 表示当 $k=0$ 时值为 1，否则为 0。最终可以从每箱中的每堆里面各拿出一些糖果放在一起，得到的快乐值为每堆糖果的快乐值的乘积（注意可以是负数）。例如 $n=2, a_1=a_2=1$ ，从第一堆里拿 0 个，第二堆里拿 3 个，获得的快乐值为 $(-1) \times (-2) = 2$ 。

对于每一种拿法，你都需要把这些糖果分给 m 个小朋友，小朋友可以不分到糖果，但是 m 个糖果必须分完。

你需要算出所有方案下，快乐值的总和是多少。两种方案不同当且仅当从某堆里拿出的糖果数量不同，或者某个小朋友分到的糖果数量不同。小朋友都是有区别的，糖果是没有区别的。

由于答案很大，你只需要输出快乐值总和 mod 998244353 的结果即可。

Round 6

Problem A (抽奖)

有一袋球，其中有 R 个红球和 B 个蓝球，初始时你手上有 X 元钱，你每次可以从手上拿出任意一部分钱（可以是任意实数）当作赌注，并猜一种颜色。然后你从这个不透明的袋子里随便拿出一个球，如果这个球的颜色和你猜的颜色一致，那么你的赌注会翻倍（即假设拿了 Y 元当赌注，那么这一轮结束之后你的钱数将变成 $X+Y$ ），否则你会失去赌注（即总钱数变成 $X-Y$ ）。

拿出的球不会再放回去，这意味着游戏会持续恰好 $R+B$ 轮。你需要找到一种策略，使得最坏情况下最终获得的钱数尽量多。注意每次拿出球之后你就知道了它的颜色（即知道袋子里还剩余多少红色和蓝色球）。

可以证明，答案一定是一个有理数。你只需要输出这个有理数在 mod 998244353 意义下对应的整数即可。

$$R+B \leq 10^3$$

一道少见的简单题。

Problem B (排列)

有 $1, 2, 3, \dots, n$ 这 n 个数排成一排，你需要解决下面两个问题：

1. 你每次可以交换任意两个数，求恰好 k 次交换后能得到多少种不同的排列。
2. 你每次可以交换相邻两个数，求恰好 k 次交换后能得到多少种不同的排列。

由于答案可能很大，你只需要输出其对 998244353 取模的结果即可。

共一行两个正整数 n, k 。

对于所有数据，保证 $1 \leq n \leq 10^9, 1 \leq k \leq 10^5$ 。

Problem C (图)

有一个 n 个点 m 条边的无向连通图，保证无重边无自环。你可以删掉原图的若干条边，然后给剩下的边赋上 $1 \sim k-1$ 的权值，使得每个点相邻的边的权值和都是 k 的倍数，且剩下的这些边可以使整张图连通。

你需要求出方案数对 998244353 取模的结果。

第二行三个正整数 n, m, k ，其中 $1 \leq n \leq 19, n-1 \leq m \leq \frac{n(n-1)}{2}, 2 \leq k \leq 3$ 。

接下来 m 行每行两个数 u, v ，表示一条边。保证无重边无自环。

Round 7

Problem A

众所周知，小葱同学擅长计算，尤其擅长计算组合数，但这个题和组合数没什么关系。

N 个大皮克敏和 M 个小皮克敏，大皮克敏有两滴血，小皮克敏有一滴血。你要开 K 枪，会等概率击中任意一只还活着的皮克敏，掉一滴血。如果最后有 a 只大皮克敏和 b 只小皮克敏还活着，则你会得到 $a \times b \times (a+b)$ 的分数。求期望分数。

$N, M \leq 2000$

Problem B

众所周知，小葱同学擅长计算，尤其擅长计算组合数，但这个题和组合数没什么关系。

定义 $F_0 = 0, F_1 = 1, F_i = F_{i-1} + F_{i-2}$ 。求 $F(F(\dots F(n))) \bmod m$ (一共 k 个 F)。

$k \leq 5, n \leq 10^{18}, m \leq 10^{18}$

这个题应该单独成文吧...

Problem C

众所周知，小葱同学擅长计算，尤其擅长计算组合数，但这个题和组合数没什么关系。

N 个数组，每个数组有 M 个 0。五种操作：

1. 给定 i, l, r, v ，将第 i 个数组的第 $l-r$ 个数增加 v 。
2. 给定 i, l, r ，询问第 i 个数组第 $l-r$ 个数的和、最小值、最大值。
3. 给定 l, r ，询问第 $l-r$ 个数组上所有数的和、最小值、最大值。
4. 给定 l, r, v ，将第 $l-r$ 个数组的所有数增加 v 。
5. 给定 i, l, r, j, t ，将第 i 个数组的第 $l-r$ 个数，移动到第 j 个数组的第 t 个数的后面。

$N, M \leq 10^5$

屑题。一道 0 思维难度的省选难度的题目。

开个线段树，每个叶子上面建动态开点 splay 就完事了。

标称 500 行。标称不压行，长得跟小葱一样

Round 8

Problem A

定义数列 $F_0 = f_0, F_1 = f_1, F_i = c_1 \times F_{i-1} + c_2 \times F_{i-2} + c_3$ 。现在定义一个长度为 n 的数组里面的元素为 F_0, F_1, \dots, F_{n-1} 。我们需要做 m 次操作，每次取出数组中前 k 个元素求和放到数组最后面去（不足 k 个取全部）。输出操作完后数组的第一个和最后一个元素。

$1 \leq n \leq 10^{18}, 0 \leq k, m \leq 10^{18}, f_0, f_1, c_1, c_2, c_3 \leq 10^9$ 。