

简要题解

sjzez czy

2019.1.23

目录

| | | |
|----------|------------------------------------------------------------|----------|
| 1 | 【51nod 1172】Partial Sums V2 | 5 |
| 1.1 | 题目大意 | 5 |
| 1.2 | 算法讨论 | 5 |
| 1.3 | 时间复杂度 | 6 |
| 1.4 | 空间复杂度 | 6 |
| 2 | 【mioj 118】Grizzly and GCD | 6 |
| 2.1 | 题目大意 | 6 |
| 2.2 | 算法讨论 | 6 |
| 2.3 | 时间复杂度 | 7 |
| 2.4 | 空间复杂度 | 7 |
| 3 | 【CCPC-Wannafly Winter Camp Day4 (Div1, onsite)】置置置换 | 7 |
| 3.1 | 题目大意 | 7 |
| 3.2 | 算法讨论 | 7 |
| 3.3 | 时间复杂度 | 7 |
| 3.4 | 空间复杂度 | 7 |
| 4 | 【CCPC-Wannafly Winter Camp Day4 (Div1, onsite)】咆哮咆哮 | 8 |
| 4.1 | 题目大意 | 8 |
| 4.2 | 算法讨论 | 8 |
| 4.3 | 时间复杂度 | 8 |
| 4.4 | 空间复杂度 | 8 |
| 5 | 【51nod 1627】瞬间移动 | 8 |
| 5.1 | 题目大意 | 8 |
| 5.2 | 算法讨论 | 9 |
| 5.3 | 时间复杂度 | 9 |

| | | |
|-----------|-----------------------------------------------------------------------------|-----------|
| 5.4 | 空间复杂度 | 9 |
| 6 | 【51nod 1149】Pi 的递推式 | 9 |
| 6.1 | 题目大意 | 9 |
| 6.2 | 算法讨论 | 9 |
| 6.3 | 时间复杂度 | 9 |
| 6.4 | 空间复杂度 | 9 |
| 7 | 【51nod 1488】帕斯卡小三角 | 10 |
| 7.1 | 题目大意 | 10 |
| 7.2 | 算法讨论 | 10 |
| 7.3 | 时间复杂度 | 10 |
| 7.4 | 空间复杂度 | 10 |
| 8 | 【bzoj 5424】烧桥计划 | 10 |
| 8.1 | 题目大意 | 10 |
| 8.2 | 算法讨论 | 11 |
| 8.3 | 时间复杂度 | 11 |
| 8.4 | 空间复杂度 | 11 |
| 9 | 【Google Code Jam 2008 APAC Onsites】Modern Art Plagiarism | 12 |
| 9.1 | 题目大意 | 12 |
| 9.2 | 算法讨论 | 12 |
| 9.3 | 时间复杂度 | 12 |
| 9.4 | 空间复杂度 | 12 |
| 10 | 【CCPC-Wannafly Winter Camp Day5 (Div1, onsite)】Special Judge | 12 |
| 10.1 | 题目大意 | 12 |
| 10.2 | 算法讨论 | 12 |
| 10.3 | 时间复杂度 | 13 |
| 10.4 | 空间复杂度 | 13 |
| 11 | 【CCPC-Wannafly Winter Camp Day5 (Div1, onsite)】Fast Kronecker Trans- | |
| | form | 13 |
| 11.1 | 题目大意 | 13 |
| 11.2 | 算法讨论 | 13 |
| 11.3 | 时间复杂度 | 13 |
| 11.4 | 空间复杂度 | 14 |

| | |
|---------------------------------------------------------------------------|-----------|
| 12 【CCPC-Wannafly Winter Camp Day5 (Div1, onsite)】 Sorting | 14 |
| 12.1 题目大意 | 14 |
| 12.2 算法讨论 | 14 |
| 12.3 时间复杂度 | 14 |
| 12.4 空间复杂度 | 14 |
| 13 【小米 OJ 编程比赛 01 月常规赛】 灯 | 15 |
| 13.1 题目大意 | 15 |
| 13.2 算法讨论 | 15 |
| 13.3 时间复杂度 | 15 |
| 13.4 空间复杂度 | 15 |
| 14 【CCPC-Wannafly Winter Camp Day8 (Div1, onsite)】 Souls-like Game | 16 |
| 14.1 题目大意 | 16 |
| 14.2 算法讨论 | 16 |
| 14.3 时间复杂度 | 16 |
| 14.4 空间复杂度 | 16 |
| 15 【CCPC-Wannafly Winter Camp Day8 (Div1, onsite)】 岸边露伴的人生经验 | 16 |
| 15.1 题目大意 | 16 |
| 15.2 算法讨论 | 16 |
| 15.3 时间复杂度 | 17 |
| 15.4 空间复杂度 | 17 |
| 16 【CCPC-Wannafly Winter Camp Day8 (Div1, onsite)】 穗乃果的考试 | 17 |
| 16.1 题目大意 | 17 |
| 16.2 算法讨论 | 18 |
| 16.3 时间复杂度 | 18 |
| 16.4 空间复杂度 | 18 |
| 17 【CCPC-Wannafly Winter Camp Day7 (Div1, onsite)】 强壮的排列 | 18 |
| 17.1 题目大意 | 18 |
| 17.2 算法讨论 | 18 |
| 17.3 时间复杂度 | 20 |
| 17.4 空间复杂度 | 20 |
| 18 【CCPC-Wannafly Winter Camp Day8 (Div1, onsite)】 玫瑰凛两开花 | 20 |
| 18.1 题目大意 | 20 |
| 18.2 算法讨论 | 21 |

| | |
|----------------------------------------------------------------------------------|-----------|
| 18.3 时间复杂度 | 21 |
| 18.4 空间复杂度 | 21 |
| 19 【NOI2009】管道取珠 | 21 |
| 19.1 题目大意 | 21 |
| 19.2 算法讨论 | 22 |
| 19.3 时间复杂度 | 22 |
| 19.4 空间复杂度 | 22 |
| 20 【POI 2007】天然气管道 | 22 |
| 20.1 题目大意 | 22 |
| 20.2 算法讨论 | 22 |
| 20.3 时间复杂度 | 22 |
| 20.4 空间复杂度 | 22 |
| 21 【2016 ACM/ICPC Asia Regional Qingdao Online 】 Herbs Gathering | 23 |
| 21.1 题目大意 | 23 |
| 21.2 算法讨论 | 23 |
| 21.3 时间复杂度 | 23 |
| 21.4 空间复杂度 | 23 |
| 22 【2018-2019 ICPC, Asia Xuzhou Regional Contest】 Rikka with Subsequences | 23 |
| 22.1 题目大意 | 23 |
| 22.2 算法讨论 | 23 |
| 22.3 时间复杂度 | 24 |
| 22.4 空间复杂度 | 24 |
| 23 【ural 2057】 Non-palindromic cutting | 24 |
| 23.1 题目大意 | 24 |
| 23.2 算法讨论 | 24 |
| 23.3 时间复杂度 | 24 |
| 23.4 空间复杂度 | 25 |
| 24 【ZJOI 2012】波浪 | 25 |
| 24.1 题目大意 | 25 |
| 24.2 算法讨论 | 25 |
| 24.3 时间复杂度 | 25 |
| 24.4 空间复杂度 | 26 |

1 【51nod 1172】Partial Sums V2

1.1 题目大意

给定一个序列（长度不超过 50000），求做 $k(k \leq 10^9)$ 次前缀和后的序列结果，序列的每个元素对 $10^9 + 7$ 取模

1.2 算法讨论

对于形式幂级数 $A(x)$ ，以及实数 p 来说，有下式成立：

$$F(x) = A(x)^p \Rightarrow A(x)F'(x) = pF(x)A'(x)$$

对于一个序列 $\{a_n\}$ 来说，求一次前缀和相当于把 $A(x)$ 变为 $\frac{A(x)}{1-x}$

若做 k 次前缀和，相当于乘上 $\frac{1}{(1-x)^k}$ ，由于乘法具有结合律，只需要考虑后式即可

令 $F(x) = \sum_{n=0}^{\infty} f_n x^n = \frac{1}{(1-x)^k}$ ，则有：

$$\begin{aligned} F(x) &= \frac{1}{(1-x)^k} = (1-x)^{-k} \\ \Rightarrow (1-x)F'(x) &= -kF(x)(-1) = kF(x) \\ \Rightarrow \sum_{n=0}^{\infty} f_{n+1}(n+1)x^n - \sum_{n=1}^{\infty} f_n n x^n &= \sum_{n=0}^{\infty} k f_n x^n \end{aligned}$$

即：

$$\begin{cases} f_0 = F(0) = 1 \\ f_{0+1}(0+1) = k f_0 \Rightarrow f_1 = k \\ f_{n+1}(n+1) - f_n n = k f_n \quad (n \geq 1) \end{cases}$$

最后一个即：

$$\begin{aligned} f_{n+1} &= \frac{(k+n)f_n}{n+1} \quad (n \geq 1) \\ \Rightarrow f_n &= \frac{(k+n-1)f_{n-1}}{n} \quad (n \geq 2) \end{aligned}$$

因此可以 $O(n)$ 计算出 $\{f_n\}$ 后，再计算 $A(x) \times F(x)$ 即可

于是问题转化为了，给定两个序列，求其卷积，其中序列长度不超过 50000，运算在模 $10^9 + 7$ 意义下

如果使用分治乘法，时间复杂度为 $O(n^{1.59})$ ，代入数据可得跑的过去

1.3 时间复杂度

$$O(n \log n)$$

1.4 空间复杂度

$$O(n)$$

2 【mioj 118】Grizzly and GCD

2.1 题目大意

给定 n, a, b , 保证 $\gcd(a, b) = 1$, 且 $1 < n, a, b < 10^5$, 求下式在模 $10^9 + 7$ 意义下的值:

$$\sum_{m=0}^n [2 \wedge \binom{n}{m}] \sum_{i=1}^n \sum_{j=1}^{i-1} \gcd(a^i - b^i, a^j - b^j)$$

2.2 算法讨论

发现这就是个二合一, 首先 $[2 \wedge \binom{n}{m}] = [n \& m = m]$, 之后考虑计算后面那个式子需要用到一个结论, 即:

$$\gcd(a, b) = 1 \Rightarrow \gcd(a^i - b^i, a^j - b^j) = a^{\gcd(i, j)} - b^{\gcd(i, j)}$$

于是就相当于求:

$$\sum_{i=1}^n \sum_{j=1}^{i-1} (a^{\gcd(i, j)} - b^{\gcd(i, j)})$$

也就是相当于求:

$$\begin{aligned} & \sum_{i=1}^n \sum_{j=1}^{i-1} a^{\gcd(i, j)} \\ &= \sum_{d=1}^n a^d \sum_{i=1}^{\lfloor \frac{n}{d} \rfloor} \sum_{j=1}^{i-1} [\gcd(i, j) = 1] \\ &= \sum_{d=1}^n a^d \left(-1 + \sum_{i=1}^{\lfloor \frac{n}{d} \rfloor} \phi(i) \right) \end{aligned}$$

直接暴力就好了

2.3 时间复杂度

$$O(n)$$

2.4 空间复杂度

$$O(n)$$

3 【CCPC-Wannafly Winter Camp Day4 (Div1, onsite)】置 置置换

3.1 题目大意

给定 n , 求有多少个 $1 \sim n$ 的全排列, 满足 $\forall 2 \leq i \leq n$, 若 $2 \mid i$, 则 $a_{i-1} > a_i$, 否则 $a_{i-1} < a_i$
其中 $1 \leq n \leq 1000$, 答案对 $10^9 + 7$ 取模

3.2 算法讨论

设 f_n 表示 $1 \sim n$ 的满足条件的全排列的个数, 则:

$$\begin{cases} f_0 = 1 \\ f_1 = 1 \\ f_n = \sum_{i=0}^{n-1} \binom{n-1}{i} f_i f_{n-1-i} [2 \nmid i] \quad (n \geq 1) \end{cases}$$

意义就是考虑第 n 个数是必须放到偶数位置上, 也就是说 n 左侧必须要有偶数个数字

3.3 时间复杂度

$$O(n^2)$$

3.4 空间复杂度

$$O(n^2)$$

4 【CCPC-Wannafly Winter Camp Day4 (Div1, onsite)】咆哮咆哮

4.1 题目大意

wls 手上有 n 张牌，每张牌他都可以选择召唤一个攻击力为 a_i 的生物，或者使得场上所有生物的攻击力加 b_i

请问如何抉择，使得场攻（场上生物攻击力的总和）最高

wls 可以任意选择出这 n 张牌的顺序

其中 $1 \leq n \leq 10^5, 0 \leq a_i, b_i \leq 10^6$

4.2 算法讨论

首先最优决策一定是先召唤若干个生物，然后一直给它们加 *buff*

设 $f(x)$ 表示召唤出 x 个生物时的最大攻击力，即有 $n - x$ 个 *buff*

由于某些原因， $f(x)$ 是一个关于 x 的单峰函数，也就是可以三分

考虑 $f(x)$ 怎么求，假设已经决定了一些卡片是召唤，一些是加 *buff*，考虑一张召唤的卡片 i 和一张加 *buff* 的卡片 j 进行交换后答案更有的条件：

$$a_j + b_i \times x > a_i + b_j \times x$$

那么按照 $a_i - a_i \times x$ 降序排序后，前 x 个卡片用于召唤，后 $n - x$ 个卡片用于加 *buff* 就好了

4.3 时间复杂度

$$O(n \log^2 n)$$

4.4 空间复杂度

$$O(n)$$

5 【51nod 1627】瞬间移动

5.1 题目大意

有一个无限大的矩形，初始时你在左上角（即第一行第一列）

每次你都可以选择一个右下方格子，并瞬移过去

求到第 $n(2 \leq n \leq 10^5)$ 行第 $m(2 \leq m \leq 10^5)$ 列的格子有几种方案，答案对 1000000007 取模

5.2 算法讨论

枚举步数 i ，之后相当于把 $n-1$ 分成 i 份， $m-1$ 分成 i 份
因此答案就是：

$$\sum_{i=1}^{\min(n-1, m-1)} \binom{n-2}{i-1} \binom{m-2}{i-1}$$

5.3 时间复杂度

$$O(n)$$

5.4 空间复杂度

$$O(n)$$

6 【51nod 1149】Pi 的递推式

6.1 题目大意

给定 $n(n \leq 10^6)$ ，求 $f(n) \bmod (10^9 + 7)$ ，其中：

$$f(n) = \begin{cases} 1 & 0 \leq n < 4 \\ f(n-1) + f(n-\pi) & n \geq 4 \end{cases}$$

6.2 算法讨论

画出转移图，考虑 f_i 对答案的贡献，枚举使用多少次 π ，然后组合数计算答案

6.3 时间复杂度

$$O(n)$$

6.4 空间复杂度

$$O(n)$$

7 【51nod 1488】帕斯卡小三角

7.1 题目大意

已知

$$\begin{cases} f_{1,j} = a_j & 1 \leq j \leq n \\ f_{i,j} = \min(f_{i-1,j}, f_{i-1,j-1}) + a_j & 2 \leq i \leq j \leq n \end{cases}$$

其中 $a(a_i \leq 10^4)$ 是一个长度为 $n(n \leq 10^5)$ 的数组

有 $m(m \leq 10^5)$ 次询问，输入 x, y ，求 $f_{x,y}$

7.2 算法讨论

手玩后发现转移单调，即一定是从第一行的某个点往下走一段距离后一直往右下方走到目标点

写出动规方程，发现是斜率优化形式

由于凸包是静态的，可以直接线段树维护区间凸包，在每个凸包上三分就行

7.3 时间复杂度

$$O(n \log^2 n)$$

7.4 空间复杂度

$$O(n \log n)$$

8 【bzoj 5424】烧桥计划

8.1 题目大意

给定一个长度为 n 的序列 a_1, a_2, \dots, a_n 和一个参数 m

你要从中删掉若干个位置 $p_1, p_2, \dots, p_k (1 \leq p_1 < p_2 < \dots < p_k \leq n)$ ，耗费 $\sum_{i=1}^k i \times a_{p_i}$ 的代价。

上一步会把序列分割成 $k+1$ 段，对于剩下的每段求和，如果某一段的和 $sum > m$ ，则要额外支付 sum 的代价

k 是你任选的，求最小总代价，其中 $n \leq 100000, 1000 \leq a_i \leq 2000$

8.2 算法讨论

考虑一个 $O(n^2)$ 的暴力, 设 $f_{i,j}$ 表示考虑完前 i 个, 第 i 个要删掉, 且一共删了 j 个的最小代价

那么有:

$$f_{i,j} = \min(f_{k,j-1} + \text{cost}(s_{i-1} - s_k)) + j \times a_i$$

其中 $\text{cost}(x) = x \times [x > m]$

转移的话可以考虑 $\text{cost}(s_{i-1} - s_k)$ 是否为 0 作为分界点

假设有一个 k , 满足: $\forall 1 \leq t \leq k, s_{i-1} - s_t > m$, 有转移:

$$f_{i,j} = \min\{f_{t,j-1} - s_t\} + j \times a_i + s_{i-1}$$

同时这个 k 应满足 $\forall k+1 \leq t \leq i-1, s_{i-1} - s_t \leq m$, 有转移:

$$f_{i,j} = \min\{f_{t,j-1}\} + j \times a_i$$

对于前半部分, 可以通过维护前缀最小值实现, 对于后半部分, 可以通过维护单调队列来实现

设 k 表示删了多少个, 对于 $k=0$ 的初始解, 它的至多代价为 $2000n$

对于一个任意 k 的解, 它的至少代价为 $\sum_{i=1}^k 1000 = 500k(k+1)$

那么如果一个 k 可能会对答案产生更优的影响, 则至少有 $500k(k+1) \leq 2000n \Rightarrow k = O(\sqrt{n})$

于是只需要保存 $k \leq O(\sqrt{n})$ 的解, 时间复杂度将为 $O(n\sqrt{n})$

8.3 时间复杂度

$$O(n\sqrt{n})$$

8.4 空间复杂度

$$O(n\sqrt{n})$$

9 【Google Code Jam 2008 APAC Onsites】Modern Art Plagiarism

9.1 题目大意

给定两棵无根树 A 和 B ，判断是否存在 A 的一个子连通块和 B 同构，其中树的节点数不超过 100

9.2 算法讨论

首先把无根树变为有根树再做，钦定 B 的根为 1，然后枚举 A 的根，使得 A 和 B 的根是同构的

如何判断两个棵树 A_r, B_r 是否同构呢？

建立二分图，如果 A_r 的某个儿子 u 和 B_r 的某个儿子 v 同构，那么连一条从 u 到 v 的边，如果这个二分图有完美匹配，那么当前这个配对节点可以同构

9.3 时间复杂度

$$O(n^2 \sqrt{n} n^2) = O(n^{4.5})$$

9.4 空间复杂度

$$O(n^2)$$

10 【CCPC-Wannafly Winter Camp Day5 (Div1, onsite)】Special Judge

10.1 题目大意

有一个 $n(1 \leq n \leq 1000)$ 个点 $m(1 \leq m \leq 2000)$ 条边的图画在了平面上，你想知道有多少对边之间对应的线段相交

特别地，对于图中的一对边，如果有公共点且只在对应的端点相交，那么我们不认为这对边相交

10.2 算法讨论

大分类讨论题

先判断是否一条线段的两个端点都在另一条线段上，之后跨立实验来判断是否可能相交，然后特判两次跨立实验的面积都为 0，然后判断是否只在某个端点处有交点

10.3 时间复杂度

$$O(m^2)$$

10.4 空间复杂度

$$O(n + m)$$

11 【CCPC-Wannafly Winter Camp Day5 (Div1, onsite)】 Fast Kronecker Transform

11.1 题目大意

给定两个序列 a_0, a_1, \dots, a_n 和 b_0, b_1, \dots, b_m ，求一个序列 c_0, c_1, \dots, c_{n+m} ，满足：

$$c_k = \sum_{i+j=k} i j \times [a_i = b_j]$$

11.2 算法讨论

枚举权值，设出现次数为 x

如果 $x \leq T$ ，那么可以直接 $O(x^2)$ 暴力卷积

否则把出现该权值的位置标为对应下标，其它位置为 0，然后进行 NTT，这个部分的时间复杂度为 $O(x \log x)$

综上，总的时间复杂度为 $O(T^2 \frac{n}{T} + n \log n \frac{n}{T}) = O(nT + n \log n \frac{n}{T})$

当 $nT = n \log n \frac{n}{T}$ 时，即 $T = \sqrt{n \log n}$ ，时间复杂度为 $O(n \sqrt{n \log n})$

实际上，由于种种原因，应该令 $T = 10^4$

11.3 时间复杂度

$$O(n \sqrt{n \log n})$$

11.4 空间复杂度

$$O(n)$$

12 【CCPC-Wannafly Winter Camp Day5 (Div1, onsite)】 Sorting

12.1 题目大意

你有一个数列 a_1, a_2, \dots, a_n ，你要模拟一个类似于快速排序的过程，同时给定一个固定的数字 x

一共有 q 次操作，诸如如下三种：

1. 询问区间 $[l, r]$ 之间的元素的和，也就是 $\sum_{i=l}^r a_i$
2. 对区间 $[l, r]$ 进行操作，也就是说你把区间中所有的数字拿出来，然后把小于等于 x 的数字按顺序放在左边，把大于 x 的数字按顺序放在右边，把这些数字接起来，放回到数列中
3. 对区间 $[l, r]$ 进行操作，也就是说你把区间中所有的数字拿出来，然后把大于 x 的数字按顺序放在左边，把小于等于 x 的数字按顺序放在右边，把这些数字接起来，放回到数列中

其中 $1 \leq n, q \leq 2 \times 10^5, 0 \leq x \leq 10^9, 1 \leq a_i \leq 10^9$

12.2 算法讨论

这道题的主要难点是在于读清楚题意，这个 x 是一个常数

那么就好做了，把 $a_i \leq x$ 的那些标为 1，把 $a_i > x$ 的那些标为 0，然后后两个操作相当于区间 01 排序，直接线段树维护区间赋值即可

由于是按照顺序重新排列，因此所有标为 0 的数字，它们在原先序列上的相对顺序不变，1 同理

然后在查询的时候只需要分别知道查询区间的 01 的个数，和之前的 01 的个数即可

12.3 时间复杂度

$$O((n + q) \log n)$$

12.4 空间复杂度

$$O(n)$$

13 【小米 OJ 编程比赛 01 月常规赛】灯

13.1 题目大意

一个屋子有 n 个开关控制着 n 盏灯，但奇怪的是，每个开关对应的不是一盏灯，而是 $n - 1$ 盏灯

每次按下这个开关，其对应的 $n - 1$ 盏灯就会由亮变灭，或者由灭变亮

保证不会有两个开关控制同样的 $n - 1$ 盏灯

现在刘同学想把灯全部开好，但是这些灯一开始的状态非常乱，刘同学想知道最少需要按多少次开关才能使所有灯全部亮起

13.2 算法讨论

这出题人水平不行啊，抄原题就算了，题面写的还不清楚

通过猜想题意，可以假设题意是这样的：有 n 个灯泡和 n 个开关，一开始编号为 $1 \sim l$ 的灯泡是亮的，第 i 个开关按下后会让所有除了第 i 号灯泡外的其它灯泡全部翻转，求最少按多少次开关可以使得所有灯全部亮起

按照套路，设 x_i 表示第 i 个开关是否按下，显然 $x_i \in \{0, 1\}$ ，且答案就是 $\sum_{i=1}^n x_i$ 的最小值同时为了满足灯泡最后都是亮的这个要求，则有如下约束：

$$\begin{cases} \oplus_{j=1 \wedge i \neq j}^n x_j = 0 & (1 \leq i \leq l) \\ \oplus_{j=1 \wedge i \neq j}^n x_j = 1 & (l+1 \leq i \leq n) \end{cases}$$

显然，只需要枚举 $T = \oplus_{i=1}^n x_i$ 的值，就会得到这个方程组的唯一解

如果 $T = 0$ ，那么 $\forall 1 \leq i \leq l, x_i = 0$ ，且 $\forall l+1 \leq i \leq n, x_i = 1$

如果 $T = 1$ ，那么 $\forall 1 \leq i \leq l, x_i = 1$ ，且 $\forall l+1 \leq i \leq n, x_i = 0$

13.3 时间复杂度

$$O(1)$$

13.4 空间复杂度

$$O(1)$$

14 【CCPC-Wannafly Winter Camp Day8 (Div1, onsite)】 Souls-like Game

14.1 题目大意

给定 n 个 3×3 的矩阵，有 m 次操作，每次诸如把 $[l, r]$ 中所有的矩阵都变为输入的一个矩阵，或者查询 $[l, r]$ 的所有矩阵的连乘积后的所有元素的和
输出对 998244353 取模，且 $n, m \leq 10^5$

14.2 算法讨论

如果直接进行线段树维护矩阵，时间复杂度是 $O(26n \log^2 n)$ 的，无法通过此题
考虑先把 n 扩充到 2^k ，这样对于每次修改，只会有 $O(\log n)$ 个参与修改的矩阵，于是可以先对于所有修改，都预处理出来，然后进行修改

14.3 时间复杂度

$$O(27(n + q) \log n)$$

14.4 空间复杂度

$$O(27(n + q \log n))$$

15 【CCPC-Wannafly Winter Camp Day8 (Div1, onsite)】 岸边露伴的人生经验

15.1 题目大意

给定 n 个 10 维向量 $\{V_n\}$ ，每一维的值在 $\{0, 1, 2\}$ 中
定义向量 $\vec{V} = (x_1, x_2, \dots, x_{10})$ 的模长 $|\vec{V}|$ 为 $\sqrt{x_1^2 + x_2^2 + x_3^2 + \dots + x_{10}^2}$
求有多少个四元组 (i, j, k, l) ，满足 $1 \leq i, j, k, l \leq n$ 且 $|\vec{V}_i - \vec{V}_j| = |\vec{V}_k - \vec{V}_l|$

15.2 算法讨论

首先去掉根号后答案不变，为了方便起见，所有的模长都平方一下
之后相当于对于所有 $0 \leq x \leq 40$ ，求有多少 (i, j) ，满足 $1 \leq i, j \leq n$ ，且 $|\vec{V}_i - \vec{V}_j|^2 = x$ ，求完之后对于所有 x ，把对数平方再求和就是答案
考虑如下的一张差的平方的真值表：

| | 0 | 1 | 2 |
|---|---|---|---|
| 0 | 0 | 1 | 4 |
| 1 | 1 | 0 | 1 |
| 2 | 4 | 1 | 0 |

再考虑异或的真值表：

| | 0 | 1 | 2 |
|---|---|---|---|
| 0 | 0 | 1 | 2 |
| 1 | 1 | 0 | 3 |
| 2 | 2 | 3 | 0 |

可以得到一个异或向差方的映射：

$$\begin{cases} 0 \Rightarrow 0 \\ 1 \Rightarrow 1 \\ 2 \Rightarrow 4 \\ 3 \Rightarrow 1 \end{cases}$$

考虑把向量 \vec{V} 看成 10 个 4 进制数，然后求异或卷积，即可得到一个异或数组，然后依次映射到差方数组即可

15.3 时间复杂度

$$O(2^{20} \times 20)$$

15.4 空间复杂度

$$O(2^{20})$$

16 【CCPC-Wannafly Winter Camp Day8 (Div1, onsite)】

穗乃果的考试

16.1 题目大意

给定一个 $n \times m$ 的 01 矩阵，设 f_i 表示恰好有 i 个 1 的子矩阵个数，求：

$$\sum_{i=0}^{nm} i^2 f_i$$

16.2 算法讨论

考虑 x^2 的等价变形:

$$x^2 = \sum_{i=1}^x \sum_{j=1}^x 1 \times 1$$

于是可以枚举两个为 1 的格子, 然后计算有多少个子矩阵包括它们
通过枚举其中一个格子后, 另一个格子通过前缀和来计算

16.3 时间复杂度

$$O(nm)$$

16.4 空间复杂度

$$O(nm)$$

17 【CCPC-Wannafly Winter Camp Day7 (Div1, onsite)】 强壮的排列

17.1 题目大意

给定奇数 n , 求有多少个 $1 \sim n$ 的全排列, 满足 $\forall 2 \leq i \leq n$, 若 $2 \mid i$, 则 $a_{i-1} < a_i$, 否则 $a_{i-1} > a_i$

其中 $1 \leq n \leq 10^5$, 答案对 998244353 取模

17.2 算法讨论

设 a_n 表示 $1 \sim n$ 的满足条件的全排列的个数, 则:

$$\begin{cases} a_0 = 1 \\ a_1 = 1 \\ a_n = \sum_{i=0}^{n-1} \binom{n-1}{i} a_i a_{n-1-i} [2 \nmid i] \quad (n \geq 1) \end{cases}$$

意义就是考虑第 n 个数是必须放到奇数位置上, 也就是说 n 左侧必须要有奇数个数字

由于输入的 n 是奇数，所以可以强行钦定 $a_{2x} = 0$ ，即所有偶数项都为 0，然后就可以去掉 $[2 \nmid i]$ 这一项了，即：

$$\begin{cases} a_0 = 0 \\ a_1 = 1 \\ a_n = \sum_{i=0}^{n-1} \binom{n-1}{i} a_i a_{n-1-i} \quad (n \geq 1) \end{cases}$$

整理可得：

$$\begin{aligned} a_n &= \sum_{i=0}^{n-1} \binom{n-1}{i} a_i a_{n-1-i} \\ \Rightarrow a_n &= \sum_{i=0}^{n-1} \frac{(n-1)!}{i!(n-1-i)!} a_i a_{n-1-i} \\ \Rightarrow n \frac{a_n}{n!} &= \sum_{i=0}^{n-1} \frac{a_i}{i!} \frac{a_{n-1-i}}{(n-1-i)!} \\ \Rightarrow n f_n &= \sum_{i=0}^{n-1} f_i f_{n-1-i} \end{aligned}$$

设 $F(x)$ 表示 $\{f_n\}$ 的生成函数，则有：

$$F'(x) = F^2(x) + 1$$

设 $y = F(x)$ ，则：

$$\begin{aligned} \frac{dy}{dx} &= y^2 + 1 \\ \Rightarrow \int \frac{dy}{y^2 + 1} &= \int 1 dx \\ \Rightarrow \arctan(y) &= x + C \\ \Rightarrow y &= \tan(x + C) \end{aligned}$$

令 $C = 0$ ，所以：

$$F(x) = \tan(x) = \frac{\sin(x)}{\cos(x)}$$

泰勒展开后多项式求逆即可

补充一个积分的推导：

设 $x = \tan y$ ，即 $y = \arctan x$

$$\begin{aligned}
& \int \frac{1}{x^2 + 1} dx \\
&= \int \frac{1}{\tan^2 y + 1} d \tan y \\
&= \int \frac{1}{\frac{\sin^2 y}{\cos^2 y} + 1} d \frac{\sin y}{\cos y} \\
&= \int \frac{\cos^2 y}{\sin^2 y + \cos^2 y} \frac{\cos^2 y - (-\sin^2 y)}{\cos^2 y} dy \\
&= y + C \\
&= C + \arctan x
\end{aligned}$$

17.3 时间复杂度

$$O(n \log n)$$

17.4 空间复杂度

$$O(n)$$

18 【CCPC-Wannafly Winter Camp Day8 (Div1, onsite)】 玖凛两开花

18.1 题目大意

给出一张点集为 V ，边集为 E 的无向图 G ，点的编号为 0 至 $|V| - 1$ ，边 (u, v) 的权值为 $\min(u, v)$

一个边集 S 是图的一个匹配当且仅当 $S \subseteq E$ ，且 $\forall e_1, e_2 \in S \wedge e_1 \neq e_2$ ，满足 e_1, e_2 无公共端点

对于一个边集 S ，定义 W_S 为 S 中所有边的权值的集合

对于一个自然数集 W ，定义 $Mex(W)$ 为最小的不属于 W 的自然数

求对于图 G 的匹配 S ， $Mex(W_S)$ 的最大值是多少

好心的 Rinne 为了减少你的负担，将题目的做法告诉你，你只需要实现一个高效的开花算法即可

当然，如果你已经会做这道题了，就可以不用继续看下去了

Rinne 给出的做法是这样的：

对于所有的边 $e \in E$ ，若其原本的边权为 w ，将其改为 $2^{|V|-w}$

求出新图的最大权匹配后，设其权值之和为 X ，将其二进制表示中的最低 $|V| + 1$ 位由高位到低位依次写出来，第一个为 0 的位的位置（从 0 开始编号）就是答案

其中 $1 \leq |V| \leq 10^4, 1 \leq |E| \leq 2 \times 10^4$

18.2 算法讨论

二分答案后，可以把原图拆分为两个二分图，只需判断是否完全匹配即可

18.3 时间复杂度

$$O(m\sqrt{n} \log n)$$

18.4 空间复杂度

$$O(n)$$

19 【NOI2009】管道取珠

19.1 题目大意

管道取珠是小 X 很喜欢的一款游戏。在本题中，我们将考虑该游戏的一个简单改版
游戏初始时，左侧上下两个管道分别有一定数量的小球（有深色球和浅色球两种类型），而右侧输出管道为空

每一次操作，可以从左侧选择一个管道，并将该管道中最右侧的球推入右边输出管道

假设上管道中有 n 个球，下管道中有 m 个球，则整个游戏过程需要进行 $n + m$ 次操作，即将所有左侧管道中的球移入输出管道

最终 $n + m$ 个球在输出管道中从右到左形成输出序列

爱好数学的小 X 知道，他共有 $\binom{n+m}{n}$ 种不同的操作方式，而不同的操作方式可能导致相同的输出序列

假设最终可能产生的不同种类的输出序列共有 K 种，其中：第 i 种输出序列的产生方式（即不同的操作方式数目）有 a_i 个

聪明的小 X 早已知道， $\sum a_i = \binom{n+m}{n}$

因此，小 X 希望计算得到： $\sum a_i^2$ ，由于这个值可能很大，因此只需要输出该值对 1024523 的取模即可

其中 $n, m \leq 500$

19.2 算法讨论

题意相当于把所有的可能的操作放到一个 *map* 中，其中相同的判定为生成的序列相同，然后输出所有相同集的大小的平方和

如果把序列复制一倍，相当于公共子序列的个数，因为在平方的时候，有序点对 (i, j) 会被计算一次

于是可以设 $f_{i,j,k,l}$ 分别表示原序列操作到了 (i, j) ，复制序列操作到了 (k, l) 时的公共子序列个数

由于 $k + l = i + j$ ，所以可以移除掉 l 这一维，然后 i 这一维可以滚动数组优化掉

19.3 时间复杂度

$$O(n^3)$$

19.4 空间复杂度

$$O(n^2)$$

20 【POI 2007】天然气管道

20.1 题目大意

平面上分别有 $n (n \leq 50000)$ 个黑点和白点，每个黑点只能像在它右下方的白点匹配
求一个黑白完美匹配，使得匹配的点对的曼哈顿距离和最小

20.2 算法讨论

若一个黑点为 (x, y) ，那么它对答案的贡献就是 $y - x$ ，白点为 $x - y$

20.3 时间复杂度

$$O(n)$$

20.4 空间复杂度

$$O(1)$$

21 【2016 ACM/ICPC Asia Regional Qingdao Online】Herbs Gathering

21.1 题目大意

给定 $n(n \leq 100)$ 个物品，每个物品有 (w, v) 两个属性，要求从中选出一些物品，使得 $\sum w$ 不超过 $m(m \leq 10^9)$ ，最大化 $\sum v$ ，保证数据随机生成

21.2 算法讨论

假设存在一个状态 $(\sum w, \sum v)$ ，如果存在另一个状态 $(\sum w', \sum v')$ ，其中 $\sum w' \leq \sum w$ 且 $\sum v' \geq \sum v$ ，那么状态 $(\sum w, \sum v)$ 就没有存在的必要了

可以证明这么做的话单次的期望状态数为 $O(\log(n!)) = O(n \log n)$

21.3 时间复杂度

$$O(n^2 \log n)$$

21.4 空间复杂度

$$O(n \log n)$$

22 【2018-2019 ICPC, Asia Xuzhou Regional Contest】Rikka with Subsequences

22.1 题目大意

给定一个长度为 $n(n \leq 200)$ 的序列 q ，每个位置是 $[1, n]$ 之间的整数

给定 $n \times n$ 的 01 矩阵 g ，定义一个序列 a_1, a_2, \dots, a_m 是好的，当且仅当且对于任意的 $1 \leq i \leq m$ ，有 $g_{a_i, a_{i+1}} = 1$ 恒成立

假设有一个 `std::map` 保存了 q 的每个好的子序列的出现次数，你需要统计它们的出现次数的立方和

22.2 算法讨论

首先对于 x^3 有一个等价变换：

$$x^3 = \sum_{i=1}^x \sum_{j=1}^x \sum_{k=1}^x 1 \times 1 \times 1$$

于是可以把序列 q 复制成三份 a, b, c , 然后求有多少个好的公共子序列

考虑第一个暴力, $f_{i,j,k}$ 表示第一个序列到了 i , 第二个序列到了 j , 第三个序列到了 k , 且强制选上 a_i, b_j, c_k 的好的公共子序列个数

转移即枚举 $i \rightarrow x, j \rightarrow y, k \rightarrow z, g_{a_i, a_x} = 1, a_x = b_y = c_z$, 时间复杂度为 $O(n^6)$

考虑优化这个暴力, 将移动分为三个阶段: 移动 i , 移动 j , 移动 k

于是可以设 $f_{i,j,k,0/1/2}$ 表示当前移动 $i/j/k$ 的方案数, 若移动 i , 则找一个 x , 使得 $g_{a_i, a_x} = 1$ 就行, 若移动 j , 则找一个 y 使得 $a_i = b_y$ 就行, k 和 j 同理, 此时时间复杂度为 $O(n^4)$

其实从枚举 x 也是没有必要的

简化转移: 要么将 x 增加 1, 要么钦定这个 x 就是我们想要的 x , 开始 y 的转移

22.3 时间复杂度

$$O(n^3)$$

22.4 空间复杂度

$$O(n^3)$$

23 【ural 2057】Non-palindromic cutting

23.1 题目大意

给定一个长度为 n 的字符串 S

将 S 划分为若干段非空连续子串, 使得每段都不是回文串

求最多能划分成多少段

23.2 算法讨论

设 f_i 表示 $s_{1\dots i}$ 的最大划分数, 则有:

$$f_i = \max(f_j) + 1$$

其中 $s_{j+1\dots i}$ 是一个有解的字符串, 即不是 $aaaaa, aabaa, ababa$ 的情况

23.3 时间复杂度

$$O(n)$$

23.4 空间复杂度

$$O(n)$$

24 【ZJOI 2012】波浪

24.1 题目大意

考虑 $1 \sim n$ 的一个排列 a_1, a_2, \dots, a_n ，定义它的波浪值为：

$$\sum_{i=1}^{n-1} |a_i - a_{i+1}|$$

给定 m ，求有多少排列的波浪值不小于 m ，其中 $n \leq 100$

24.2 算法讨论

考虑从小到大的顺序依次插入 $1 \sim n$ ，设 $f_{i,j,k,l}$ 表示现在要插入 i ，已经有了 j 个极大连续块，此时的贡献和为 k ，有 l 个边界被碰触了

对于绝对值符号，可以拆为：

$$|a - b| = \max(a, b) - \min(a, b)$$

考虑如下几种转移：

1. 放在两侧，不和任何已有的极大连续块相邻

此时的方案数为 $2 - l$ ，且有贡献为 $-i$

2. 放在两侧，和某个极大连续块相邻

此时的方案数为 $2 - l$ ，且有贡献为 i

3. 合并两个已存在的极大连续块

此时的方案数为 $j - 1$ ，且有贡献为 $2i$

4. 独自成为一个极大连续块

此时方案数为 $j + 1 - l$ ，且有贡献 $-2i$

5. 只和一个已存在的极大连续块相邻

此时方案数为 $2j - l$ ，且有贡献 $i - i = 0$

24.3 时间复杂度

$$O(n^4)$$

24.4 空间复杂度

$$O(n^3)$$