

基础算法

搜索、分治、贪心、暴力和构造

2019年7月15日

黄哲威 hzwer



自我介绍

- 北京大学16级计算机科学方向
- 计算概论 A, 数据结构与算法 A , 算法设计与分析讨论班助教
- NOI银牌, CTSC金牌, ACM区域赛金牌
- 旷视科技(Megvii) Research Intern 研究计算机视觉与强化学习
- hzwer.com

本节目标

- 基本的枚举（搜索）算法，二分算法，分治算法，贪心算法

回顾经典模型

通过刷题锻炼识别题目类型的能力

- 介绍一些竞赛政策和经验

前置技能 - 简单数据结构

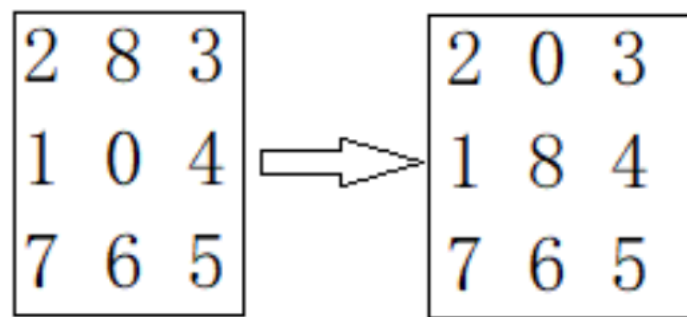
- 二叉堆，可以在 $O(\log n)$ 的时间内插入删除， $O(1)$ 的时间查询最小值，要求掌握 `priority_queue`
- 平衡树， $O(\log n)$ 插入删除， $O(\log n)$ 查询一个元素的前驱后继，要求掌握 `set` 和 `multiset`，也可以用于查询最大最小值
- `map` 可以当一个哈希表，使用形如一个下标范围扩大的数组
- 参考 <https://wenku.baidu.com/view/93f33b3b192e45361066f5eb.html>

搜索

- 深度搜索和广度搜索，优先选择比较好实现的深度搜索。
- 一般只在网格图寻路，分层图 dp 的时候使用广搜。
- 特别注意：不要用深度搜索求最短路！
- 做好复杂度估计，直接回溯通不过可以考虑折半搜索或者剪枝。
- 如果有必要，用 hash（推荐 map）存储已经搜过的状态。

经典模型. 八数码

- 在 3×3 的棋盘上，摆有八个棋子，每个棋子上标有 1 至 8 的某一数字。棋盘中留有一个空格，空格用 0 来表示。
- 空格周围的棋子可以移到空格中。
- 要求解的问题是：给出一种初始布局（初始状态）和目标布局（为了使题目简单,设目标状态为 123804765），找到一种最少步骤的移动方法，实现从初始布局到目标布局的转变。



状态1

状态2

例题.买汽水

- 暑假有 N 天，总共有 M 块钱，每天可以选择花 p_i 的钱买汽水，也可以不买。问最多能花掉多少钱？
- 30%的数据， $N \leq 20$
- 100%的数据， $N \leq 40$, $p_i \leq 10^9$, $M \leq 10^9$

例题.买汽水

- 首先这个问题看起来很像 01 背包的 NPC，没有特殊性质的话，解法一定是搜索！
- 30%的数据， $N \leq 20$ ， 2^N 回溯 2^{20} 有 6 个 0，可以通过
- 100%的数据， $N \leq 40$ ， $p_i \leq 10^9$ ， $M \leq 10^9$
 - $N = 40$ 启发我们使用折半搜索
 - 把前半的方案 $\{x\}$ 搜出来存在一个数组里排序，对于后半的每一个方案 y ，在 $\{x\}$ 中二分查找 $M - y$ 的前驱
 - 也可以用平衡树 (set) 维护

分治

- 分成相同或相似的子问题, 子问题可简单的直接求解, 原问题的解即子问题的解的合并
- 注意复杂度计算, 考虑分治每一层的开销, 不要把问题想复杂
- 复杂度分析:
- 递归树: $T(n) = kT(n/m) + f(n)$

对于归并排序: $T(n) = 2T(n/2) + O(n) \Rightarrow O(n \log n)$

$T(n) = T(n/2) + O(n) \Rightarrow O(n)$

$T(n) = 2T(n/2) + O(n^2) \Rightarrow O(n^2)$

例题.逆序对

- 设 A 为一个有 n 个数字的数列，其中所有数字各不相同。
- 如果存在正整数 i, j 使得 $1 \leq i < j \leq n$ 而且 $A[i] > A[j]$ ，则这一个有序对称为 A 的一个逆序对。逆序对的数量称作逆序数。
- 求一个数列的逆序数？ $n \leq 10^5$

例题.逆序对

- 树状数组 or 平衡树，即按顺序考虑每个 A_i ，在它之前有多少个比它大的数，略
- 归并排序
 - 1.划分问题：把序列分成元素个数尽量相等的两半
 - 2.递归求解：把两半元素分别排序
 - 3.合并问题：把两个有序表 $O(n)$ 合并为一个，双指针

求逆序对只要考虑两个有序表合并的时候新产生多少逆序对

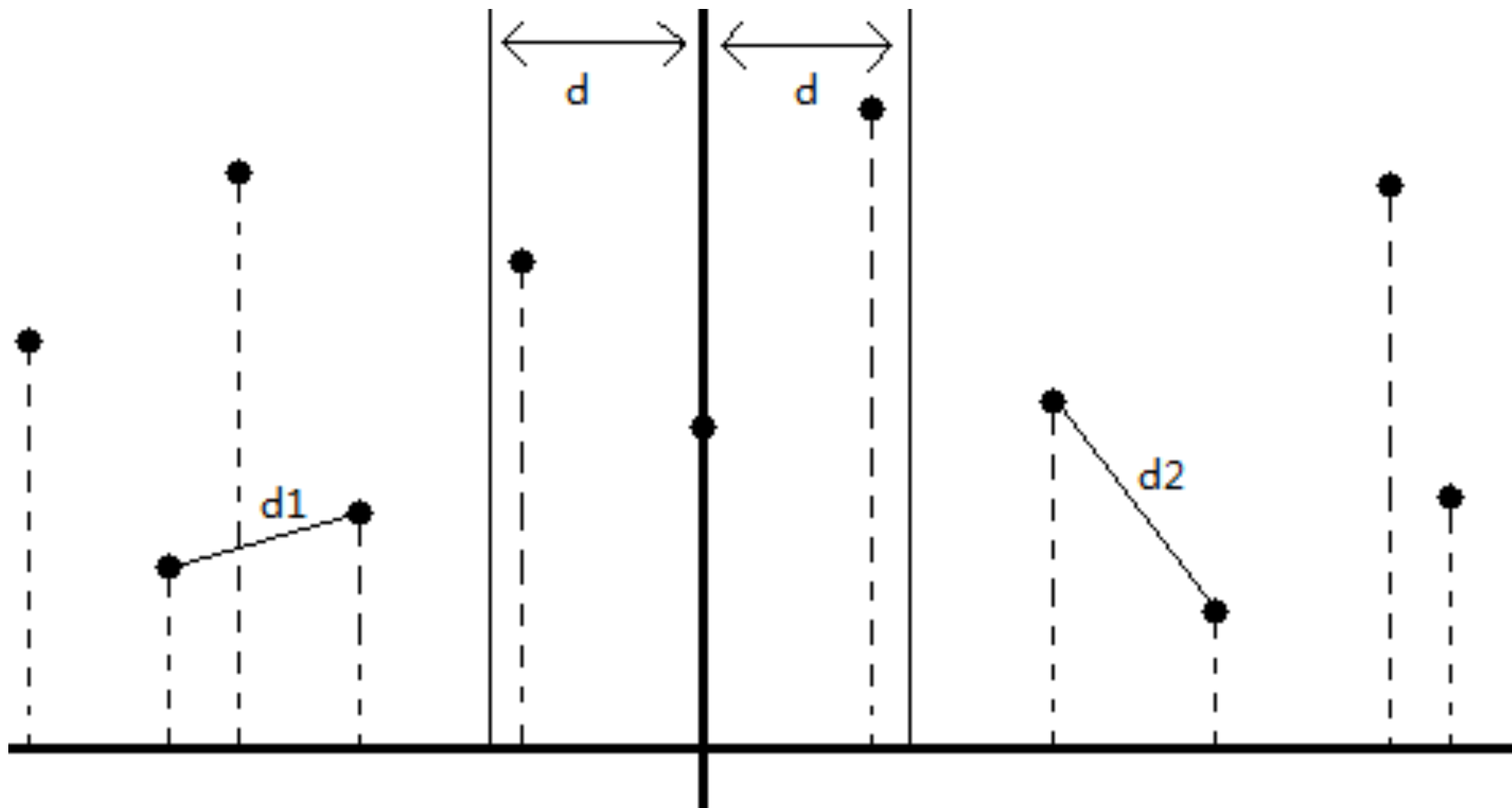
例题.最近点对

- n 个点在公共空间中，求出所有点对中，欧几里得距离最小的点对。
- $n \leq 10^5$

例题.最近点对

- 分解：对所有的点按照 x 坐标（或者 y ）从小到大排序，根据下标进行分割，使得点集一分为二，分为两个集合。
- 解决：分别求出两个子集的最近点对值——子问题之间相互独立
- 子问题的（最优）解可以合并得到全局的最优解：但最近距离不一定存在于两个集合中，可能一个点在集合A，一个点在集合B，而这两点间距离小于 dis 。
- 怎么考虑跨越中线的最近点对？

例题.最近点对



练习题

CF140C.New Year Snowmen

- 要堆起一个雪人，需要三个不同大小的雪球
- 现在有 n 个雪球，第 i 个雪球大小是 a_i 个，问最多能堆起多少个雪人，给定的雪球大小可能相同
- $1 \leq n \leq 10^5, 1 \leq a_i \leq 10^9$

CF140C.New Year Snowmen

- 数量少的雪球更“珍贵”, 数量最多的雪球先用
- 计算每种大小的雪球的个数, 并放入堆中
- 每次取出前三多的雪球大小, 用这三种雪球堆一个雪人

CF175C. Geometry Horse

- 有 n 种数字，第 i 种数字的数量为 k_i ，价值为 c_i
- 依次取走数字，取走某个数字时获得的价值为 $c_i * f$
- 其中 f 表示当前系数，初始为1
- 当取走第 p_i 个数字后， $f += 1$ ， p_i 表示第 i 个素数
- 可以以任意顺序取走数字，求可获得的最大价值和
- $n \leq 100$ ， $k_i \leq 100$

CF175C. Geometry Horse

- 当前系数 f 单调不降
- 越后取的数字，乘的当前系数 f 越大
- 以数字大小排序所有数字，按价值从小到大取走数字，计算答案

CF486B.OR in Matrix

- A, B 都是 $n*m$ 的 01 矩阵，已知 B 矩阵是由 A 矩阵以一种规则生成
- B_{ij} 是由 A 矩阵的第 i 行的所有元素和第 j 列的所有元素进行或运算 得到
- 给定 B 矩阵，求是否存在一个矩阵 A 能生成 B
- $1 \leq n, m \leq 100$

CF486B.OR in Matrix

- 如果矩阵B的某个值为 0，则 A 矩阵整行整列都是 0
- 如果矩阵B的某个值为 1，那 A 矩阵行或列上至少有 1 个 1
- 先填 0，其余都填 1，最后验证一下

CF724B.Batch Sort

- 给一个 $n*m$ 的矩阵，每一行是一个 $1-m$ 的全排列
- 可以交换矩阵的两列，然后每一行还可以交换两个元素
- 问能不能使得每一行都是单调递增的
- $1 \leq n, m \leq 300$

CF724B.Batch Sort

- 暴力枚举要交换的两列，然后逐行判断是否能够通过一次交换使得一整行单调递增

CF460C.Present

- 有 n 盆花，编号 1 到 n ，每盆花都有高度值 h_i
- 浇 m 次水，每次只能浇连续的 w 盆花，每浇一次，被浇的花高度值 $+1$ 。
- 希望让其中最矮的花最高，问浇完水后最矮的花的高度的最大值？
- $1 \leq n, m, w \leq 10^5, 1 \leq h_i \leq 10^9$

CF460C.Present

- 二分最优解，贪心判定
- 从左到右，如果某盆花小于二分值，将其以及后面的 w 盆花 $+1$
- 用线段树 / 差分 + 前缀和维护

CF482A.Diverse Permutation

- 构造 n 的一个全排列，使其相邻数之间的差值有 K 种，若不存在输出 No
- 例：1 5 4 2 3 的差值序列是 4 1 2 1，有 3 种差值
- $1 \leq n, k \leq 10^5$

CF482A.Diverse Permutation

- $1 \sim n$ 最多凑出 $n - 1$ 种差值
- 8 1 7 2 6 3 5 4 的差值序列是 7 6 5 4 3 2 1
- 构造 k 项差值为 2 到 k , 其余的差值全为 1 即可

CF425A.Sereja and Swaps

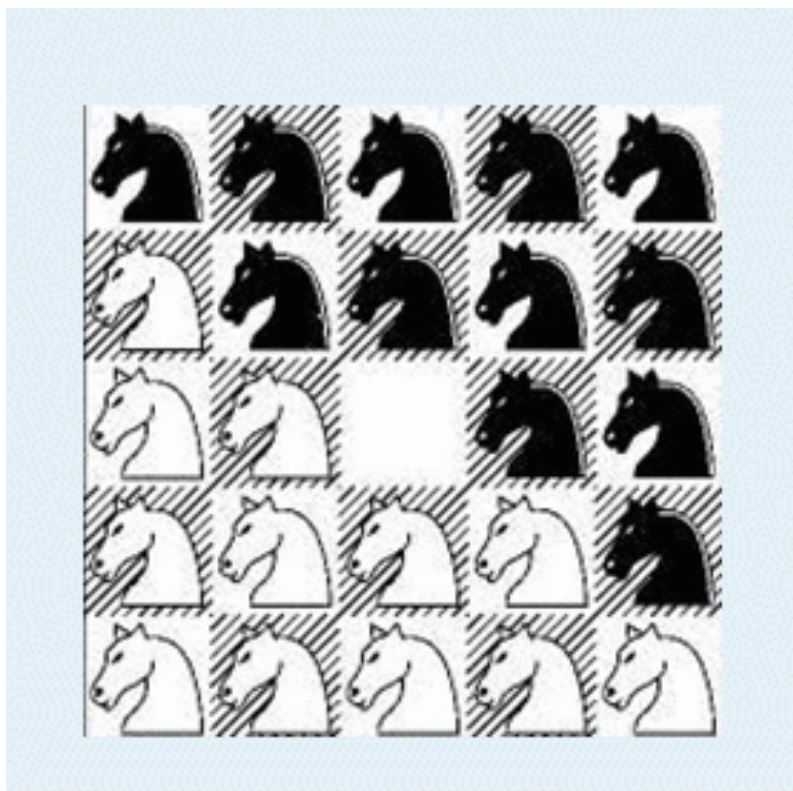
- 给一个长为 n 的序列，以及交换次数 k
- 每次可以在原先的序列中任意交换两个数
- 交换后找一个最大子串和，输出其可能的最大值。
- $1 \leq n \leq 200, 1 \leq k \leq 10$

CF425A.Sereja and Swaps

- 枚举一个子串，将子串内最小的和子串外最大的尝试交换

BZOJ1085. 骑士精神

- 在一个 5×5 的棋盘上有12个白色的骑士和12个黑色的骑士， 且有一个空位。在任何时候一个骑士都能按照骑士的走法（它可以走到和它横坐标相差为1，纵坐标相差为2或者横坐标相差为2，纵坐标相差为1的格子）移动到空位上。
- 给定一个初始的棋盘，怎样才能经过移动变成如下目标棋盘，求最少的步数（保证步数 ≤ 15 ）



BZOJ1085. 骑士精神

- 双向广搜，正着搜 7 步，反着搜 8 步
- 迭代加深搜索 + 剪枝
- A* 估计当前至少还要跳几步，如果一定得不到最优解则剪枝

Fibonacci-ish

- 给一个长为($n \leq 1000$)的数列, $1 \leq \text{数字} \leq 10^9$
- 问最多能选出多少个数组成广义斐波那契数列
- 广义斐波那契数列数列定义为: 从数列的第三项开始, 每一项等于前两项的和

Fibonacci-ish

- 枚举选出两个作为斐波那契数列的开头，暴力递推
- 由于斐波那契数列的增长速度很快，几十项就会超过 10^9 ，所以复杂度是 $O(n^2)$ 的
- 增长到底有多快呢？

LOJ6559. 小奇采药

- n 个药材，每一个有价值 v ，采摘时间 t
- 问总时间 m 的情况下最多能获得多大价值
- 保证 t, v 在限制范围内均匀随机生成

对于 30% 数据, $1 \leq n \leq 20, 1 \leq m, v_i, t_i \leq 10^4$;

对于 60% 数据, $1 \leq n \leq 100, 1 \leq m, v_i, t_i \leq 10^5$;

对于 100% 数据, $1 \leq T \leq 10, 1 \leq n \leq 150, 1 \leq m, v_i, t_i \leq 10^9$ 。

LOJ6559. 小奇采药

- 对于 30% 的数据， $O(2^n)$ 枚举取 or 不取
- 对于 60% 的数据， $O(nm)$ 做 01 背包，即 $f(i, j)$ 表示前 i 株草药，耗费 j 的时间能达到的最大代价。
- 对于 100% 的数据，注意到 m, t, v 纯随机，那么不会选太多的草药，而耗时较少的草药有很大概率存在于最优解中

LOJ6559. 小奇采药

- 针对这些性质优化搜索：
 1. 排序，把采摘时间长的排在前面
 2. 如果当前剩下的时间不够采最后一个，剪枝
 3. 如果把剩下的都拿了都不会更新当前解，剪枝
- 当然也可以合理使用随机化和卡时，复杂度 $O(\text{玄学})$

LOJ6560. 小奇取石子

- 有 n 堆石子，第 i 堆石子有 a_i 个，最多取 k 堆石子，请问在要求总石子数不超过 k 的情况下最多能取多少石子。

数据分为 A、B、C 三组，各占 30%、30%、40%；

对于 A 组数据， $1 \leq m \leq n \leq 10, 1 \leq k \leq 1000, 1 \leq a_i \leq 100$ ；

对于 B 组数据， $1 \leq m \leq n \leq 20, 1 \leq k \leq 10^8, 1 \leq a_i \leq 10^6$ ；

对于 C 组数据， $1 \leq m \leq n \leq 200, 1 \leq k \leq 2500, 1 \leq a_i \leq 50$ ；

LOJ6560. 小奇取石子

- 在 n 个数字当中取出 m 个，使得总和小于 k 。这是一个比较显然的背包动态规划模型，复杂度是 $O(kmn)$
- 当然，对于 n 较小的数据，我们甚至可以使用 $O(2^n)$ 的算法暴力搜索每一种匹配方式，然后判断是否符合条件。
- 同时，我们注意到B组数据的 k 范围比较大，如果采用背包dp算法时间复杂度太高，我们也开不出那么大的数组来记录状态。
- 但是B组数据的 n 太小了，采用暴力搜索反而能过。所以，我们对于不同组的数据需要不同处理。

LOJ6560. 小奇取石子

数据分为 A、B、C 三组，各占 30%、30%、40%；

对于 A 组数据， $1 \leq m \leq n \leq 10, 1 \leq k \leq 1000, 1 \leq a_i \leq 100$ ；

对于 B 组数据， $1 \leq m \leq n \leq 20, 1 \leq k \leq 10^8, 1 \leq a_i \leq 10^6$ ；

对于 C 组数据， $1 \leq m \leq n \leq 200, 1 \leq k \leq 2500, 1 \leq a_i \leq 50$ ；

- 对于A组数据，我们注意到n和m的数据范围很小，只有10，那么直接暴力搜索或者做背包动态规划都能通过。
- 对于B组数据，由于k的范围过大，我们只能暴力搜索。
- 对于C组数据，由于n的范围过大，我们只能背包dp。

CF1187C.Vasya And Array

- 构造一个长为 n 的序列 A ，满足 m 个限制条件
- 限制条件有两种
 1. 要求 $[l_i, r_i]$ 这一段序列是严格递增的
 2. 要求 $[l_i, r_i]$ 这一段序列不是严格递增的
- 问满足条件的序列是否存在？如果存在则输出任一合法序列。
- $1 \leq n \leq 1000$, $1 \leq m \leq 1000$, 要求 $0 \leq A_i \leq 10^9$

CF1187C.Vasya And Array

- 考虑差分后的序列，初始值为全 0
- 如果要求一段是严格递增的，则这一段全设成 1
- 把所有严格递增的先满足，再考虑每个非严格递增的，在这些区间内至少找一个设成 -1

POI2005. Toy Cars

- 有 n 个不同的玩具，接下来 p 个时刻每次要玩其中的一个玩具
- 地上可以放 k 个玩具，如果一个要玩的玩具不在地上，就要到架子上去拿，地上的玩具超过 k 的时候要选择一个放回
- 求从架子上拿玩具的最小次数
- $k, n \leq 10^5, p \leq 5 * 10^5$

POI2005. Toy Cars

- 一开始肯定把前 K 个要玩的玩具放地上，考虑放回哪个玩具
- 记录下每个玩具下次出现的时间，每次要放回的话选一个离现在时刻最远的玩具
- 用堆维护这一过程

CF767E. Change Free

- 现在有面值为100的纸币和1元的硬币，纸币无限多，但是硬币只有 m 个。
- 接下来 n 天，每天都要去食堂花费 $c[i]$ 元。
- 已知收银员在第 i 天找零 x 元的话，不满意度会增加 $x * w[i]$
- 求最小的不满意度之和，并输出方案（每天用几张纸币和几个硬币）
- $n \leq 10^5$, $m \leq 10^9$

CF767E. Change Free

- 只用考虑 $c[i] \% 100$ 的部分。
- 我们可以每天都用硬币，直到第 k 天发现硬币不够了。那就说明我们必须在 $1 \sim k$ 天中的某一天找一次零。
- 假设我们选择第 i 天 ($1 \leq i \leq k$)，本来是花了 $c[i]$ 个硬币。
- 现在改成使用纸币，节省下了 $c[i]$ 个硬币，还找回了 $(100 - c[i])$ 个硬币。
- 因此无论 $c[i]$ 多少，都会多出 100 个硬币。

CF767E. Change Free

- 选择第 i 天的代价是 $w[i](100-c[i])$
- 因此，如果第 k 天硬币不够了，那么就在 $1 \sim k$ 天中选一个代价最小的，兑换 100 个硬币即可。
- 用最小堆来维护代价。
- 这样就保证每一步决策都是最优的，贪心策略正确！

CF1190B. Tokitsukaze, CSL and Stone Game

- n 堆石子，每堆有 a_i 个，两个人玩取石子游戏。
- 两人轮流，每次取一个。如果一个人没有石子可以取，或者取完石子后，存在两堆一样多的石子，则失败。
- 一堆的石子数可以是 0，不能是负数。
- 两人都遵循最优策略，问先手必胜还是后手必胜？
- $1 \leq n \leq 10^5, 0 \leq a_i \leq 10^9$

CF1190B. Tokitsukaze, CSL and Stone Game

- 排序不影响答案。
- 如果初始就有两堆一样的石子，先手必须取一个，如果取完以后还有两堆一样的，则后手胜利。
- 接下来可以直接计算总共能取多少次石子，即把第 i 小的石子变成 i (从 0 开始标号)。以总操作数的奇偶性判断胜负。

CF196C. Paint Tree

- 给定二维平面上的 N 个点，保证没有三点共线。
- 给定一棵树，请在二维平面上的 N 个点之间连 $N-1$ 条线，使得这些线无交且与给定的树同构。
- 数据保证有解，输出任意方案。
- $N \leq 1500$

CF196C. Paint Tree

- 选择树中的任意一点作为根，用 dfs 求出每个子树的大小
- 选二维平面中最左边的点为根
- 对其他点做极角排序，按照极角序分配给每个子树，分治处理
- 无三点共线→必然有解

codechef. CLPERM

- 数字 1 - N 中丢失了 K 个数字，求：不能用剩余数字加和得到的数字中 最小的一个 的奇偶性，数字可以重复用
- 如果所有自然数都能凑出，输出 -1
- $1 \leq N \leq 10^9, 1 \leq K \leq 10^6$

codechef. CLPERM

- 设最小的未丢失数字是 mn
- 如果当前凑出了 $1 \sim x$, 则 $1 \sim x + mn$ 都可以凑出来
- 依次考虑 $1 \sim (mn - 1)$ 能不能凑出
- 用已有数字从小到大做筛法
- 由调和级数, 得出复杂度是 $O(K \log K)$

POJ Challenge. 生日礼物

- 给一个序列 A ，选择不超过 M 个连续的部分，求最大值
- $N \leq 10^5$, $M \leq 10^5$, $A \leq 10^9$

POJ Challenge. 生日礼物

- 首先将符号相同的一段合并成一个值，为了处理方便一开始可以将序列中的 0 全部去掉，记为序列 B
- 如 0 1 3 -1 4 -1 -3 变成 4 -1 4 -3，可以证明不影响最终结果
- 不考虑负数的情况：将 B 中所有 > 0 的元素放到小根堆里，若有 K 个，取出最小的 $K - m$ 个舍弃
- 考虑上负数：把负数取反后放到堆里，从堆中取出一个负元素的意思是，把它和左右两个一起作为一个连续的部分
- 当取出一个元素的时候，将其与在序列 B 的前后元素合并，把合并后的结果的绝对值加入堆