南渝国庆训练1

恰饭

算法一

当 $a_1 = a_2 = a_3 = a_4, b_1 = b_2 = b_3$ 时，每一种菜和甜品的组合的花费都是相同的，输出 $a_1 + b_1$ 即可。

时间复杂度 $O(1)$，期望得分 20 分。

算法二

当 $a_1 = a_2 = a_3 = a_4$ 时，每一种菜的价格是相同的，枚举选哪个甜品，取个最小值即可。

时间复杂度 $O(1)$，期望得分 40 分。

算法三

最优解肯定是价格最少的菜和价格最少的甜品，答案即为 $\min\limits_{i=1}^{4} a_i + \min\limits_{j=1}^{3} b_j$。

时间复杂度 $O(1)$，期望得分 100 分。

卡片

算法一

当 $k = 1$ 时，答案即为卡片上不同数字个数。

时间复杂度 $O(n)$，期望得分 20 分。

算法二

当 $a_1 = a_2 = \ldots = a_n$ 时，答案为 1。

时间复杂度 $O(1)$，期望得分 20 分。

算法三

暴力搜索选出了哪些卡片，将卡片上的数字拼起来，存到一个数组中，最后寻找数组中总共有多少元素即可。

时间复杂度 $O\left(\binom{n}{k} \cdot k!\right)$，期望得分 100 分。

数数

算法一

暴力枚举每个点是否在 S 中，再暴力判断是否合法，

时间复杂度 $O(2^n \cdot n \log n)$，期望得分 8 分。

算法二

按照所有点 y 降序排序，令 $f_{l,i,j}$ 表示到第 l 个点，选出的集合最后两个元素为 x_i, x_j 的方案数，可以枚举上一个 k 转移。

时间复杂度 $O(n^3)$，空间复杂度 $O(n^3)$，期望得分 20 分。

算法三

发现算法二可以滚动数组优化。

时间复杂度 $O(n^3)$，空间复杂度 $O(n^2)$，期望得分 36 分。

算法四

发现算法三还可以前缀和优化。

时间复杂度 $O(n^2)$，空间复杂度 $O(n^2)$，期望得分 100 分。

算法五

更换思路，按照所有点 x 升序排序。

考虑将 x 作为横坐标，y 作为纵坐标，从后往前选点，选出的点 y 一定是单调递增的，且一定是 x 坐标向左/向右交替进行选点。

将所有点按照 x 排序，令 f_i, g_i 表示填完的最后一位是 i，接下来向左/向右填的方案数。

考虑转移：

对于 g_j，只需要枚举下一个填的位置 i 满足 $y_i \geq y_j$，令 $f_i = f_i + g_j$。

对于 f_j，我们可以每次填入两个位置，我们需要找到下一个向左的位置 i 满足 $i \geq j$ 转移即可。

具体的说对于一个 i，我们可以倒序枚举 j：

- 如果 $y_i > y_j$：$f_i \leftarrow g_j$。
- 如果 $y_i < y_j$：$g_j \leftarrow f_i$。

时间复杂度 $O(n^2)$，空间复杂度 $O(n)$，期望得分 100 分。

划分

算法一

暴力枚举每个节点和它的哪个儿子在一条直链上（或者和它的任意一个儿子都不在一条直链上），判断每条直链是否合法。

时间复杂度 $O(2^n \cdot n)$，期望得分 30 分。

算法二

对于 $a_1, a_2, \ldots, a_n$ 互不相同，$b_1, b_2, \ldots, b_n$ 互不相同的情况，每个权值最终到达的节点都是唯一的，源点和到达点一定是在同一条直链上，否则不合法。如果某两条链之间有交且一个不包含另一个也不合法，反之一定合法。

时间复杂度 $O(n)$，期望得分 20 分。

算法三

考虑一种构造方法，我们每次选择一个叶子 u。考虑从 u 开始不断向父亲跳，直到当前的直链满足条件。若跳到根还是不合法或者该过程中存在一个点已经被划分到了另一条直链上，那么一定无解（因为其他划分方案都是在这种方案的基础上将某些链合并而成的）。否则将该直链加入划分方案。

时间复杂度 $O(n)$，期望得分 100 分。

南渝国庆训练2

第一题 优美的数

知识点

枚举，预处理

做法

由于给了样例$k = 2021$

我们可以对优美的数进行预处理，得到前2021个优美的数。然后读入k并输出答案即可。 总时间复杂度$\mathcal{O}(k)$。

第二题 异或序列

知识点

位运算，二分, 哈希

做法

对于前50%的点，我们考虑使用一个$O(n^2)$ 的二重循环，枚举所有的数对，判断是否为答案。

对于接下来25%的数据，我们考虑$a\ xor\ b = c$ 等价于 $a\ xor\ c = b$.那么对于每一个a_i 都有一个固定的数 $ans = a_i\ xor\ X$.我们用$cnt[i]$记录数字i的出现次数，即可统计答案。时间复杂度$O(n + v)$. Tips : 此做法可以借助std::map 扩展到100%的情况,但是要取决于常数问题(由于常数问题，导致某不愿透露姓名的出题人挂了3个点)。

对于全部的数据，考虑排序数组后，每一个a_i对应的ans都是一个连续的区间。我们二分出左右端点，即可在$O(n \log_2 n)$的时间内统计答案。

第三题 分！身！术！

知识点

单调队列, two-pointer

做法

对于30%的数据，我们考虑枚举所有的区间，然后分别统计两人做题时间，判断是否可行，时间复杂度$\mathcal{O}(n^3)$\ 对于60%数据，我们在枚举区间的同时，使用ST表$\mathcal{O}(1)$统计两人的做题时间，判断合法性。时间复杂度$\mathcal{O}(n^2)$\ 对于80%的数据，我们可以用"Two-Pointer"做法，用set数据结构，维护当前区间答案，并且可以在$\mathcal{O}(\log n)$的时间进行转移，总复杂度达到了$\mathcal{O}(n\log n)$\ 对于100%的数据，使用two-pointer在移动区间右端点时，使用单调队列，对a_i, b_i分别维护最大值,最小值。当差值超过K时，记录答案，并移动区间左端点位置。时间复杂度$\mathcal{O}(n)$

第四题 种树

知识点

树形dp, 换根dp

做法

对于45%的数据，枚举受力点进行$O(n^2)$的计算即可。

对于100%的数据： 首先，假设受力点为1并计算此时的做功W_1，时间复杂度$\mathcal{O}(n)$

然后，我们考虑将受力点从u转移到v（u和v之间有连边），可以得到对于v的子树中的所有点i，dis_i将减小1，对于v的子树以外的所有点j，dis_j将增加1。由此可知，
$ans_v = ans_u - \sum_{i在v的子树内} w_i + \sum_{j不在v的子树内} w_j = ans_u + \sum_{i=1}^{n} w_i - 2\sum_{i在v的子树内} w_i$。在预处理所有点的子树重量和的情况下，可以在时间复杂度$\mathcal{O}(1)$内由ans_u得到ans_v。

总时间复杂度$\mathcal{O}(n)$。