

生产工厂 (factory)

签到题，先排序，接下来可以选择二分答案，但是二分需要注意中间值如果使用乘法的话，会爆 long long，所以使用整除比较大小即可。

或者排完序直接计算答案也可以，处理每个前缀一共需要加工多少个零件以及有多少时间，除一除算个最大值即可。

树的深度 (tree)

设 $f_{n,m}$ 表示 n 个点这样生成的树，最大深度为 m 的方案数是多少。

考虑枚举 2 号节点子树大小，剔除 2 号节点子树后的仍然是一个子问题，所以 dp 转移如下：

$$f_{n,\max(i,j+1)} \leftarrow f_{n-k,i} \times f_{k,j} \times \binom{n-2}{k-1}$$

直接转移是 $O(n^4)$ 的，分类讨论一下 $i > j$ 和 $i \leq j$ 的情况，使用前缀和优化即可做到 $O(n^3)$ 。

优美序列(sequence)

还是一个 dp，暴力 dp 是这样的： $f_{i,j}$ 表示 i 选了，上一个选的是 j 的最长优美子序列长度，转移如下：

$$f_{i,j} = \max_{k < j \wedge a_k \neq a_i} \{f_{j,k} + 1\}, a_i \neq a_j$$

时间复杂度是 $O(n^3)$ 的，无法接受。

考虑优化，容易发现，每次查询相当于把 $f_{j,*}$ 中的和 a_i 相等给去除了，所以我们只要求出 $f_{j,*}$ 的最大值以及 a 与之不同的次大值即可。

所以新的状态是 $f_i = (s, t), g_i = (u, v)$ ，表示 i 选了，最长优美子序列长度为 s ，上一个选的元素 a 的值为 t ，上一个选的 a 的值不为 t 的最长优美子序列长度为 u ，上一个选的是 v 。

考虑如何转移，我们发现，如果两个选择的数中间，出现了至少五个不同的数，那么一定不优（这五个中至少有一个是左右两边各两个没有出现过的，可以直接多选择一个），所以我们只需要从 i 开始向前枚举最近的五个不同的数，从前面的 f, g 转移而来即可。

时间复杂度 $O(n)$ 。

博弈游戏 (game)

比较有意思的一个数位 dp 题。

设 $f_{i,s,j}$ 表示一个数的最后 i 位都是 0，更高位的和为 s ，这个数往前（不算本身）有恰好 j 个先手必胜的状态，这个数加 10^i 之后往前（不算本身）有恰好多少个先手必胜的状态。

转移如下：

$$f_{i,s,j} = f_{i-1,s+9,\dots f_{i-1,s+1,f_{i-1,s,j}}}$$

边界条件：

$$f_{0,s,j} = \begin{cases} 0 & s \leq j \\ j + 1 & s > j \end{cases}$$

每次查询，也直接从 0 开始跳，跳到 $n + 1$ 为止，就能求出 n 开始往前有几个先手必胜状态，接下来那一个就是先手必败状态。

这样的时间复杂度为 $O(10^2 \log^3 n + 10T \log n)$ ，能够通过。

更进一步也能优化，利用先手必败状态必然是 10 的倍数，可以把个位直接去掉（认为是 0，答案最后再加），然后 j 这一维就一定是 10 的倍数减一了，于是可以直接记录其加一除以 10 的结果，那么时间复杂度优化为 $O(10 \log^3 n + 10T \log n)$ 。