题目选讲：动态规划 & 数论

清华大学 任舍予

2024 年 8 月 8 日

问题 (CSP-S 2021 括号序列)

定义超级括号序列为所有合法括号序列中插入若干段长度不超过 k 的 $*$ 所得到的序列。
给定 n, k 与由 $(,), *, ?$ 构成的字符串 S，求有多少种将 $?$ 替换为 $(,), *$ 的方案，使得得到的序列是超级括号序列。
数据范围：$1 \le k \le n \le 500$。

- 按照得到合法括号序列的方式可以设计区间 DP。

- 按照得到合法括号序列的方式可以设计区间 DP。
- 直接转移的问题在于：对于多段合法括号序列的拼接，可能会被重复计算。

- 按照得到合法括号序列的方式可以设计区间 DP。
- 直接转移的问题在于：对于多段合法括号序列的拼接，可能会被重复计算。
- 因此 DP 时分别记录所有的情况和不可拆分（即最外层一定是一对匹配的括号）的情况对应的方案数即可。

- 按照得到合法括号序列的方式可以设计区间 DP。
- 直接转移的问题在于：对于多段合法括号序列的拼接，可能会被重复计算。
- 因此 DP 时分别记录所有的情况和不可拆分（即最外层一定是一对匹配的括号）的情况对应的方案数即可。
- 转移时需要前缀和优化。

- 按照得到合法括号序列的方式可以设计区间 DP。
- 直接转移的问题在于：对于多段合法括号序列的拼接，可能会被重复计算。
- 因此 DP 时分别记录所有的情况和不可拆分（即最外层一定是一对匹配的括号）的情况对应的方案数即可。
- 转移时需要前缀和优化。
- 时间复杂度 $O(n^3)$，空间复杂度 $O(n^2)$。

问题 (NOIP 2021 数列)

给定 n, m, k 与权值序列 $v_0, v_1, \ldots, v_m$。
定义序列 $[a_1, \ldots, a_n]$ 合法当且仅当 $0 \le a_i \le m$ 且 $S = \sum_{i=1}^{n} 2^{a_i}$ 的二进制表示中 1 的个数不超过 k。
定义序列 $[a_1, \ldots, a_n]$ 的权值为 $\prod_{i=1}^{n} v_{a_i}$。
求所有合法序列的权值和。
数据范围：$1 \le k \le n \le 30$，$0 \le m \le 100$。

- 由于限制与二进制表示相关，考虑类似于数位 DP 的方法。

- 由于限制与二进制表示相关，考虑类似于数位 DP 的方法。
- 为了方便考虑进位的贡献，从低位到高位进行 DP：记 $f_{i,j,k,p}$ 表示包含 j 个 $0 \sim i$，且当前数位和为 k，需要进位 p 的序列个数。

- 由于限制与二进制表示相关，考虑类似于数位 DP 的方法。
- 为了方便考虑进位的贡献，从低位到高位进行 DP：记 $f_{i,j,k,p}$ 表示包含 j 个 $0 \sim i$，且当前数位和为 k，需要进位 p 的序列个数。
- 转移时只需要枚举当前一位的个数，然后直接乘组合数统计方案。

- 由于限制与二进制表示相关，考虑类似于数位 DP 的方法。
- 为了方便考虑进位的贡献，从低位到高位进行 DP：记 $f_{i,j,k,p}$ 表示包含 j 个 $0 \sim i$，且当前数位和为 k，需要进位 p 的序列个数。
- 转移时只需要枚举当前一位的个数，然后直接乘组合数统计方案。
- 时间复杂度 $O(n^3km)$，空间复杂度 $O(n^2km)$。

问题 (NOIP 2021 方差)

给定非严格递增的序列 $[a_1, \ldots, a_n]$。每次可以进行的操作是：任意选择一个正整数 $1 < i < n$，将 a_i 变为 $a_{i-1} + a_{i+1} - a_i$。求在若干次操作之后，该数列的方差最小值。

数据范围：$1 \le n \le 10^4$，$1 \le a_1 \le \cdots \le a_n \le 600$。

- 注意到每次进行的操作类似于关于 a_{i-1} 与 a_{i+1} 的中点对称翻转，因此不能发现操作的本质为交换差分序列的相邻两项。

- 注意到每次进行的操作类似于关于 a_{i-1} 与 a_{i+1} 的中点对称翻转，因此不能发现操作的本质为交换差分序列的相邻两项。
- 容易证明，方差取到最小值时，差分序列一定单谷。因此考虑从中间往两侧 DP，即按从小到大的顺序插入所有差分值。

- 注意到每次进行的操作类似于关于 a_{i-1} 与 a_{i+1} 的中点对称翻转，因此不能发现操作的本质为交换差分序列的相邻两项。
- 容易证明，方差取到最小值时，差分序列一定单谷。因此考虑从中间往两侧 DP，即按从小到大的顺序插入所有差分值。
- 直接用差分序列计算方差比较困难，可以还原到原序列计算，于是 DP 时需要记录当前的序列和，时间复杂度为 $O(n^2a)$。

- 注意到每次进行的操作类似于关于 a_{i-1} 与 a_{i+1} 的中点对称翻转，因此不能发现操作的本质为交换差分序列的相邻两项。
- 容易证明，方差取到最小值时，差分序列一定单谷。因此考虑从中间往两侧 DP，即按从小到大的顺序插入所有差分值。
- 直接用差分序列计算方差比较困难，可以还原到原序列计算，于是 DP 时需要记录当前的序列和，时间复杂度为 $O(n^2a)$。
- 注意到差分数组之和为 $\max a$，因此非零的数个数有限，只需要对这部分进行 DP 即可。

- 注意到每次进行的操作类似于关于 a_{i-1} 与 a_{i+1} 的中点对称翻转，因此不能发现操作的本质为交换差分序列的相邻两项。
- 容易证明，方差取到最小值时，差分序列一定单谷。因此考虑从中间往两侧 DP，即按从小到大的顺序插入所有差分值。
- 直接用差分序列计算方差比较困难，可以还原到原序列计算，于是 DP 时需要记录当前的序列和，时间复杂度为 $O(n^2a)$。
- 注意到差分数组之和为 $\max a$，因此非零的数个数有限，只需要对这部分进行 DP 即可。
- 时间复杂度 $O(na\min(n,a))$，空间复杂度 $O(na)$。

问题 (NOIP 2023 天天爱打卡)

有 n 天，每天可以选择跑步或不跑，跑步会使能量 $-d$。连续跑步不能超过 k 天。

给定 m 个区间 $[l,r]$，每个区间有权值 w，如果这个区间中的日子均选择跑步，则能量 $+w$。

求 n 天后能量的最大值。

数据范围：$1 \le k \le n \le 10^9$，$1 \le m \le 10^5$。

- 首先不难得到一个按日期进行 DP 的做法，每次枚举一段跑步的连续天数然后转移。

- 首先不难得到一个按日期进行 DP 的做法，每次枚举一段跑步的连续天数然后转移。
- 一方面，转移只需要从每个区间的端点处出发，因此可以离散化后只计算这些位置的 DP 值。

- 首先不难得到一个按日期进行 DP 的做法，每次枚举一段跑步的连续天数然后转移。
- 一方面，转移只需要从每个区间的端点处出发，因此可以离散化后只计算这些位置的 DP 值。
- 另一方面，可以转移的位置是一个后缀，可以用双指针计算出可行的转移点。同时，转移的代价为子区间和，于是可以直接使用线段树进行优化。

- 首先不难得到一个按日期进行 DP 的做法，每次枚举一段跑步的连续天数然后转移。
- 一方面，转移只需要从每个区间的端点处出发，因此可以离散化后只计算这些位置的 DP 值。
- 另一方面，可以转移的位置是一个后缀，可以用双指针计算出可行的转移点。同时，转移的代价为子区间和，于是可以直接使用线段树进行优化。
- 时间复杂度 $O(m\log m)$，空间复杂度 $O(m)$。

问题 (洛谷 P8564)

给定一棵 n 个点的有根树，根节点为 1。
给定权值序列 $[f_2, f_3, \ldots, f_n]$。一次删除操作为选择一个结点，然后删去其子树内除它以外的所有点，并产生 f_s 的代价，其中 s 为选择结点的当前子树大小。
求若干次删除操作后删去除 1 以外所有点的最小代价。
数据范围：$1 \leq n \leq 5000$。

- 设 $f_{i,j}$ 表示在 i 的子树中删除 j 个结点的最小代价。

- 设 $f_{i,j}$ 表示在 i 的子树中删除 j 个结点的最小代价。
- 转移时需要将每个儿子删去的部分相加，满足树上背包的形式，因此可以直接计算。

- 设 $f_{i,j}$ 表示在 i 的子树中删除 j 个结点的最小代价。
- 转移时需要将每个儿子删去的部分相加，满足树上背包的形式，因此可以直接计算。
- 时间复杂度 $O(n^2)$，空间复杂度 $O(n^2)$。

问题 (CF1515E)

有 n 台电脑排成一排。每次可以手动开启一台。若某台电脑未开启且它两侧的电脑均开启，则它会自动开启。求开启所有电脑的方案数。两种方案不同当且仅当对应的手动开启电脑的序列不同。

数据范围：$3 \leq n \leq 400$。

- 不难发现开启的方式一定为开启若干个距离为 1 的连续段。

- 不难发现开启的方式一定为开启若干个距离为 1 的连续段。
- 对于一个长度为 l 的连续段，其开启方法为 2^{l-1} 种。

- 不难发现开启的方式一定为开启若干个距离为 1 的连续段。
- 对于一个长度为 l 的连续段，其开启方法为 2^{l-1} 种。
- 直接按顺序 DP，记录之前的手动开启的数量即可乘组合数转移。

- 不难发现开启的方式一定为开启若干个距离为 1 的连续段。
- 对于一个长度为 l 的连续段，其开启方法为 2^{l-1} 种。
- 直接按顺序 DP，记录之前的手动开启的数量即可乘组合数转移。
- 时间复杂度 $O(n^3)$，空间复杂度 $O(n^2)$。

问题 (CF1572C)

给定颜色序列 $[a_1, \ldots, a_n]$，每次操作可以选定一个相等的同色段改为另一种颜色，求最少的操作次数使得整个序列颜色相同。
数据范围：$1 \le n \le 3000$。保证每种颜色初始时只有不超过 20 个。

- 注意到一定不会修改两个相交但不包含的区间，于是可以区间 DP。

- 注意到一定不会修改两个相交但不包含的区间，于是可以区间 DP。
- 不难证明，对于区间 $[l, r]$，将 $a_l \sim a_r$ 全部改为 a_l 总是不劣的。

- 注意到一定不会修改两个相交但不包含的区间，于是可以区间 DP。
- 不难证明，对于区间 $[l, r]$，将 $a_l \sim a_r$ 全部改为 a_l 总是不劣的。
- 同时，如果拆成两段分别改成同色，除非改的颜色相同，要不然划分的方式是没有区别的。

- 注意到一定不会修改两个相交但不包含的区间，于是可以区间 DP。
- 不难证明，对于区间 $[l, r]$，将 $a_l \sim a_r$ 全部改为 a_l 总是不劣的。
- 同时，如果拆成两段分别改成同色，除非改的颜色相同，要不然划分的方式是没有区别的。
- 于是转移只有 $[l+1, r]$ 与满足 $c_l = c_k$ 的 $[l, k-1], [k, r]$ 两种。

- 注意到一定不会修改两个相交但不包含的区间，于是可以区间 DP。
- 不难证明，对于区间 $[l, r]$，将 $a_l \sim a_r$ 全部改为 a_l 总是不劣的。
- 同时，如果拆成两段分别改成同色，除非改的颜色相同，要不然划分的方式是没有区别的。
- 于是转移只有 $[l+1, r]$ 与满足 $c_l = c_k$ 的 $[l, k-1], [k, r]$ 两种。
- 时间复杂度 $O(n^2)$，空间复杂度 $O(n^2)$。

问题

给定 m 组限制 (a,b)，求满足以下条件的合法的 $1 \sim n$ 的出栈序列个数：

- 对于任意一组限制 (a,b)，a 不能在 b 之后出栈。

数据范围：$3 \leq n \leq 300$。

- 一个排列 $[p_1, \ldots, p_n]$ 是合法的出栈序列，当且仅当：
  - 不存在 $1 \leq i < j < k \leq n$，使得 $p_i > p_k > p_j$。

- 一个排列 $[p_1, \ldots, p_n]$ 是合法的出栈序列，当且仅当：
  - 不存在 $1 \le i < j < k \le n$，使得 $p_i > p_k > p_j$。
- 考虑 p_n 的限制，实际上要求 $[1, p_n - 1]$ 中的数一定要比 $[p_n + 1, n]$ 中的数先出栈。

- 一个排列 $[p_1, \ldots, p_n]$ 是合法的出栈序列，当且仅当：
  - 不存在 $1 \le i < j < k \le n$，使得 $p_i > p_k > p_j$。
- 考虑 p_n 的限制，实际上要求 $[1, p_n - 1]$ 中的数一定要比 $[p_n + 1, n]$ 中的数先出栈。
- 于是可以按权值进行区间 DP：记 $f_{l,r}$ 表示将 $l \sim r$ 出栈的合法出栈序列个数，通过枚举区间中最后一个出栈的数转移。

- 一个排列 $[p_1, \ldots, p_n]$ 是合法的出栈序列，当且仅当：
  - 不存在 $1 \leq i < j < k \leq n$，使得 $p_i > p_k > p_j$。
- 考虑 p_n 的限制，实际上要求 $[1, p_n - 1]$ 中的数一定要比 $[p_n + 1, n]$ 中的数先出栈。
- 于是可以按权值进行区间 DP：记 $f_{l,r}$ 表示将 $l \sim r$ 出栈的合法出栈序列个数，通过枚举区间中最后一个出栈的数转移。
- 对于给定的 m 组限制，对每次转移的限制均满足二维数点的形式，预处理二维前缀和即可快速判断。

- 一个排列 $[p_1,...,p_n]$ 是合法的出栈序列，当且仅当：
  - 不存在 $1 \le i < j < k \le n$，使得 $p_i > p_k > p_j$。
- 考虑 p_n 的限制，实际上要求 $[1,p_n-1]$ 中的数一定要比 $[p_n+1,n]$ 中的数先出栈。
- 于是可以按权值进行区间 DP：记 $f_{l,r}$ 表示将 $l \sim r$ 出栈的合法出栈序列个数，通过枚举区间中最后一个出栈的数转移。
- 对于给定的 m 组限制，对每次转移的限制均满足二维数点的形式，预处理二维前缀和即可快速判断。
- 时间复杂度 $O(n^3)$，空间复杂度 $O(n^2)$。

问题

给定一棵带权无根树，点权为 $0/1/2$。求断掉若干条边后，点权和为 k 的连通块个数的最大值。

数据范围：$1 \leq n \leq 10^6$。

- 每次选择一个存在点权和为 k 的连通块的极小子树，将其划分出去一定是最优的。

- 每次选择一个存在点权和为 k 的连通块的极小子树，将其划分出去一定是最优的。
- 对于一个点权和为 k 的极小连通块：

- 每次选择一个存在点权和为 k 的连通块的极小子树，将其划分出去一定是最优的。
- 对于一个点权和为 k 的极小连通块：
  - 若至少有一个叶子点权为 2，则可以删去它得到点权和为 $k-2$ 的连通块；

- 每次选择一个存在点权和为 k 的连通块的极小子树，将其划分出去一定是最优的。
- 对于一个点权和为 k 的极小连通块：
  - 若至少有一个叶子点权为 2，则可以删去它得到点权和为 $k-2$ 的连通块；
  - 若所有叶子点权都为 1，则可以删去两个叶子得到点权和为 $k-2$ 的连通块。

- 每次选择一个存在点权和为 k 的连通块的极小子树，将其划分出去一定是最优的。
- 对于一个点权和为 k 的极小连通块：
  - 若至少有一个叶子点权为 2，则可以删去它得到点权和为 $k-2$ 的连通块；
  - 若所有叶子点权都为 1，则可以删去两个叶子得到点权和为 $k-2$ 的连通块。
- 于是只需要求出点权和分别为奇数/偶数的最大连通块即可判断出是否存在点权和为 k 的连通块，这可以通过树形 DP 解决。

- 每次选择一个存在点权和为 k 的连通块的极小子树，将其划分出去一定是最优的。
- 对于一个点权和为 k 的极小连通块：
  - 若至少有一个叶子点权为 2，则可以删去它得到点权和为 $k-2$ 的连通块；
  - 若所有叶子点权都为 1，则可以删去两个叶子得到点权和为 $k-2$ 的连通块。
- 于是只需要求出点权和分别为奇数/偶数的最大连通块即可判断出是否存在点权和为 k 的连通块，这可以通过树形 DP 解决。
- 时间复杂度 $O(n)$，空间复杂度 $O(n)$。

问题 (CF1801F)

给定 n, k 与序列 $[a_1, \ldots, a_n]$，求满足 $\prod_{i=1}^{n} b_i \geq k$ 的正整数序列 $[b_1, \ldots, b_n]$ 的 $\prod_{i=1}^{n} \left\lfloor \frac{a_i}{b_i} \right\rfloor$ 的最大值。
数据范围：$1 \leq n \leq 100$，$1 \leq k \leq 10^7$，$1 \leq a_i \leq 10^7$。

- 设计 DP 如下：$f_{i,j}$ 表示 $\prod_{l=i+1}^{n} b_l \geq j$ 时 $\prod_{l=1}^{i} \left\lfloor \frac{a_l}{b_l} \right\rfloor$ 的最大值。

- 设计 DP 如下：$f_{i,j}$ 表示 $\prod_{l=i+1}^{n} b_l \geq j$ 时 $\prod_{l=1}^{i} \left\lfloor \frac{a_l}{b_l} \right\rfloor$ 的最大值。
- 注意到每次转移是从 j 转移到 $\lceil j/x \rceil$，而 $\lceil j/x \rceil$ 的取值只有 $O(\sqrt{k})$ 种。

- 设计 DP 如下：$f_{i,j}$ 表示 $\prod_{l=i+1}^{n} b_l \geq j$ 时 $\prod_{l=1}^{i} \left\lfloor \frac{a_l}{b_l} \right\rfloor$ 的最大值。
- 注意到每次转移是从 j 转移到 $\lceil j/x \rceil$，而 $\lceil j/x \rceil$ 的取值只有 $O(\sqrt{k})$ 种。
- 于是每次转移时直接枚举不同的 j 和 $\lceil j/x \rceil$，积分得到复杂度是 $O(k^{3/4})$ 的。

- 设计 DP 如下：$f_{i,j}$ 表示 $\prod_{l=i+1}^{n} b_l \ge j$ 时 $\prod_{l=1}^{i} \left\lfloor \frac{a_l}{b_l} \right\rfloor$ 的最大值。
- 注意到每次转移是从 j 转移到 $\lceil j/x \rceil$，而 $\lceil j/x \rceil$ 的取值只有 $O(\sqrt{k})$ 种。
- 于是每次转移时直接枚举不同的 j 和 $\lceil j/x \rceil$，积分得到复杂度是 $O(k^{3/4})$ 的。
- 时间复杂度 $O(nk^{3/4})$，空间复杂度 $O(n+\sqrt{k})$。