

# 联合省选 2022 讲评

墨染空

2022.5.13

- ① D1T1. 预处理器
- ② D1T2. 填树
- ③ D1T3. 学术社区
- ④ D2T1. 卡牌
- ⑤ D2T2. 序列变换
- ⑥ D2T3. 最大权独立集问题

## ① D1T1. 预处理器

## ② D1T2. 填树

## ③ D1T3. 学术社区

## ④ D2T1. 卡牌

## ⑤ D2T2. 序列变换

## ⑥ D2T3. 最大权独立集问题

## 题解

- 模拟。递归节点数可能是  $1000 * 100 * 100$ ，那只要把每个字符串编一个号， $O(1)$  递归就行了。

## 反思与总结

- 怎么保证模拟不写挂？
- 复杂度的预估？为什么出了考场才发现复杂度假了？

① D1T1. 预处理器

② D1T2. 填树

③ D1T3. 学术社区

④ D2T1. 卡牌

⑤ D2T2. 序列变换

⑥ D2T3. 最大权独立集问题

- ① 考虑枚举最小值，按现在值分成的区间分成  $O(n)$  段，每段答案是多项式，对于每段，就是要求这个区间都代入求和。（对于第二问，相当于仅仅是多加一维，记录当前有没有选择一条边，这样意义相当于是枚举一条边的贡献，求经过他的路径总数）
- ② 如何获取多项式方法 1：枚举最小值，过程中再记录一维表示有没有节点的选择得到了最小值。
- ③ 如何获取多项式方法 2：使用长度为  $k$  的区间减去  $k-1$  区间即可。
- ④ 代入求和思路 1：前缀和后插值就行。
- ⑤ 代入求和思路 2：不需要插值，直接暴力维护多项式，使用  $O(n^2)$  的多项式乘法，复杂度是树上背包的，仍不变。然后求等幂和代入即可。
- ⑥  $O(n^3)$

## 反思与总结

- ① 比赛时如何有信心写出来？
- ② 有巨大多 bug 怎么办？
- ③ 怎么保证不挂分？



① D1T1. 预处理器

② D1T2. 填树

③ D1T3. 学术社区

④ D2T1. 卡牌

⑤ D2T2. 序列变换

⑥ D2T3. 最大权独立集问题

## 题解

- 没有使用欧拉回路，我们先考虑先解决 C 部分分。
- 我们先给出一个假设，之后给出构造证明：有一个有向图，每个点代表一个信息， $x \rightarrow y$  表示  $x$  要恰好在  $y$  前面，保证每个点出入度不超过 1。这样的图，一定能构造出一条对应的等价链，满足符合情况的尽可能多。
- 如果以上结论是正确的，那么最优解很好使用一个二分图匹配模型描述：左边是一个点尝试找出度（或者表示上面），右边是一个点尝试找入度（下面），两者有边当且仅当能带来 1 的贡献。
- 答案就是最大匹配数，建图可以建虚点优化成  $2n$  点， $4m$  边。

## 题解

- 考虑任意构造一组方案，此时是若干链和环，考虑把环相当于是都是同向，那么任意一个地方断开，然后嵌入学术社区（一定在链上）的那个位置，肯定可以做到的，于是完成了构造。
- 以 loushang 的环为例，找到环上一个任意位置  $A \rightarrow B$ ，然后  $B$  到  $A$  的中间部分合称  $D$ ，现在  $B$  类型学术没限制的称为  $B'$ ，现在有状态  $C \rightarrow B'$ ，那变成  $C \rightarrow B \rightarrow D \rightarrow A \rightarrow A$ ，没有任何匹配消失，把环嵌入了链里。
- 考虑加入双向匹配，我们再给出一个结论，所有双向匹配直接匹配肯定不劣，就是说一定是极大的。现在考虑我们一开始那个有向图，假设有一对双向匹配  $AB$ ，现在状况是  $Ay$ ， $xB$  和匹配和  $AB$ ， $xy$ ，那前者想要严格强，至少得有三个匹配信息，也就是至少有一个是双向匹配，那意味着要么  $A, x$  限制完全相同，或者  $y, B$  限制完全相同，那这个决策就是等价的。

## 题解

- 复杂度分析同多路增广的二分图匹配，即增广路长度  $\leq \sqrt{m}$  的时候只需要增广  $\leq \sqrt{m}$  次，增广路长度  $> \sqrt{m}$  的时候最多有  $\sqrt{m}$  条不相交的增广路，于是复杂度  $O(m\sqrt{m})$

## 反思与总结

- 如果想到了，但很不确定对不对，又不会证明，怎么办？
- 写了过不去大样例，应该如何决策？
- 分配时间？
- 这种题怎么入手？
- 碰到魔怔题，如果避免受到魔怔背景的影响？

- ① D1T1. 预处理器
- ② D1T2. 填树
- ③ D1T3. 学术社区
- ④ D2T1. 卡牌
- ⑤ D2T2. 序列变换
- ⑥ D2T3. 最大权独立集问题

# 题解

- 对值域根号分治，按大于根号的质数对质数分组。
- 每次钦定一个质数集合容斥，就是选的数一个都不能是有这个质数集合里的，那么枚举小的质数集合，每个大的是一个数乘起来。
- 预处理每个大组钦定小组一个集合的贡献，就做完了。
- 预处理部分可以暴力 for，我考场暴力 for 的  $O(2^{14} \times 14 \times n)$ ，但好像很危险，差点爆零了，更快的方法是 OR 卷积，大概是 OR 卷积。当然整个过程可以用 FWT 来推到  $O(2^{14} \times 14 \times \frac{n}{\ln n})$ 。
- $O(2^{14} \sum c)$

- 这题怎么拍？
- 如果没有做过寿司晚宴，会很难想到根号分治吗？为什么根号分治在这个结构上有优异？
- 其实 OR、AND 卷积用组合意义容斥解释很容易证明，但 XOR 卷积那个 xor 的 popcount 奇偶性好像没有直观感受？所以 xor 卷积 FWT 比较优异，没有其他做法？（或者是隐式容斥？）



- ① D1T1. 预处理器
- ② D1T2. 填树
- ③ D1T3. 学术社区
- ④ D2T1. 卡牌
- ⑤ D2T2. 序列变换**
- ⑥ D2T3. 最大权独立集问题

## 题解

- 考虑变成括号树，每次找一个留着，剩下推到下一层。这样肯定不亏。

# 题解

- $x = 0, y = 1$
- 每次往下推是自己的权值!!! 那么每次保留最大的权值不往下推就好了!

# 题解

- $x = 1, y = 1$
- 往下推是自己权值，再加上现在还在这层的随便选一个，考虑这层现在有  $t$ ，考虑钦定推下去的  $t-1$  个，前  $t-2$  个都可以靠着最小值推，然后最后一个可以靠着最小值推。那么所以每次推权值都是一个加上最小值，并且你可以把最大的留下，当前贡献和对之后的贡献都是优秀的！所以贪心就好啦！

## 题解

- $x = 1, y = 0$
- 设  $t_i$  为第  $i$  层需要往下推的个数，这个对于所有过程都是一样的！每次往下推是自己的权值！那么每次保留最大的权值不往下推就好了！
- 考虑  $t$  长这样：1 1 1 2 2 2 2 3 3 4 4 ... 5 4 3 2 1
- $b_i$  是留下的权值， $mn$  是这层最小的权值！
- 你整体的看，这个  $\sum b$  相当于是所有权值 - 最后一层的权值！
- 所以你要让最后一层权值尽量大，然后推下去的尽量小。

## 题解

- $x = 1, y = 0$
- 然后考虑如果  $t_i$  大于等于 3 那你留下次大值一定不亏啦!
- 只有  $t_i$  等于 2 的那个连续段有问题, 后面的 321 没有问题, 2 的时候你留下最小值, 肯定对的!
- 考虑那个 2 连续段, 就是你要选一个扔下去。考虑之后的过程都是贪心, 设扔下去的这个值为  $x$ , 考虑任意一种方案, 答案都可以表示为关于  $x$  的一次函数, 那么这时候想要最优  $x$  要么最大值最小值, 所以扔下去的一定是 2 连续段中的最大或者最小, 分别做一下就做完了!

## 反思总结

- 考试主要想不到的是从小往大推，和这个 - 最后一层权值，说明我们抓住局部的同时，需要整体看！
- 括号问题经常需要转化成别结构，更好描述？
- 如何讨论清楚？
- 如何保证正确性？这道题怎么拍？
- 这种题该如何训练？

- ① D1T1. 预处理器
- ② D1T2. 填树
- ③ D1T3. 学术社区
- ④ D2T1. 卡牌
- ⑤ D2T2. 序列变换
- ⑥ D2T3. 最大权独立集问题



## 题解

- 这个题和树上的数十分类似，我一开始思路就是确定一个点出度边的顺序，那么谁去谁的就确定了。并且这个顺序只要求知道每个点的相邻边关系，整个谁去谁都是固定的！
- 所以可以 DP！你交换父亲边的时候，会把一个子树的点送上去，一个子树外的点挖下来，那这时候你不知道子树外的点去哪了，所以这个需要记录。
- 他记录的是一个考虑  $u$  子树的边，一个从父亲边上去走啊走，一个人从父亲边下来走啊走，两方面的贡献都需要计算。
- 状态即  $F_{u,x,y}$ （假装  $u$  不是根，根特判），表示  $u$  父亲边换的时候，把子树内的  $x$  换上去了，换下来的点是  $y$ 。但是无论怎样，状态都是  $O(n^3)$

## 题解

- 这时候一个神秘的方法：将记录谁下来改为记录下来的点最终去哪了，这里记得两个东西都是在子树内的，并且肯定分属两个不同子树或者是自己，这样空间就可以  $O(n^2)$  了，并且仍然可以转移。
- 你考虑这个 DP 过程每个点的花费如何被统计的，在往上走的时候，一步步更新贡献，往下走的时候，在往下走之前的那个位置（即路径的 LCA 处）直接加上这个距离的贡献。
- 总结一下，状态即  $F_{u,x,y}$ （假装  $u$  不是根，根特判），表示  $u$  父亲边换的时候，把子树内的  $x$  换上去了，换下来的点最后去到了  $y$ 。其中一定满足  $LCA(x,y) = u$ ，所以不用记第一维。
- DP 通过大分讨一个点的边贡献，大概有七种。通过最简单的预处理等，可以优化成  $O(n^2)$ 。

# 题解

- Case 0:  $u$  是叶子。
- $f_{u,u} = 0$ 。

# 题解

- Case 1.1:  $u$  有一个儿子  $v$ , 先换儿子边再换父亲边。
- 枚举儿子的状态  $f_{A,B}$  它加上  $w_A + w_u \times (d_B - d_u)$  会贡献到  $f_{A,u}$  (这里  $w$  是权值,  $d$  是深度)
- 新增贡献分别代表:  $w_A$  上来一步,  $w_u$  推下去
- $O(n^2)$

# 题解

- Case 1.2:  $u$  有一个儿子  $v$ , 先换父亲边再换儿子边。
- 枚举儿子的状态  $f_{A,B}$  它加上  $w_A$  会贡献到  $f_{u,B}$
- 新增贡献分别代表:  $w_A$  上来一步。
- $O(n^2)$

# 题解

- Case 2.1: 有两个儿子, 分别是  $x, y$ , 换边顺序是  $x, y, fa_x$
- 枚举  $x$  的状态  $f_{A,B}$ ,  $y$  的状态  $f_{C,E}$  它们加上  $w_A \times (d_E - d_u + 1) + w_u \times (d_B - d_u) + w_C$  会贡献到  $f_{C,u}$
- 新增贡献分别代表:  $w_A$  上来一步然后走到  $E$ ,  $u$  往下走到  $B$ ,  $C$  上来一步。
- 发现除了  $A, E$  剩下贡献两个都是分离的, 那就先预处理每个  $A$  对应的最优的值, 然后同时枚举  $A, E$  贡献到  $E$  对应最优的贡献, 然后枚举  $C, E$  更新  $dp$  数组。  $O(n^2)$

## 题解

- Case 2.2: 有两个儿子, 分别是  $x, y$ , 换边顺序是  $x, fa_x, y$
- 枚举  $x$  的状态  $f_{A,B}$ ,  $y$  的状态  $f_{C,E}$  它们加上  $w_A + w_u \times (d_B - d_u) + w_C$  会贡献到  $f_{A,E}$
- 新增贡献分别代表:  $w_A$  上来一步,  $u$  往下走到  $B, C$  上来一步。
- 全是分离的, 那就先预处理每个  $A, E$  对应的最优的值, 然后同时枚举  $A, E$  贡献到 dp 数组。  $O(n^2)$

# 题解

- Case 2.3: 有两个儿子, 分别是  $x, y$ , 换边顺序是  $fa_x, x, y$
- 枚举  $x$  的状态  $f_{A,B}$ ,  $y$  的状态  $f_{C,E}$  它们加上  $w_A \times (d_E - d_u + 1) + w_C$  会贡献到  $f_{u,B}$
- 新增贡献分别代表:  $w_A$  上来一步然后走到  $E$ ,  $C$  上来一步。
- 发现除了  $A, E$  剩下贡献两个都是分离的, 那就先预处理每个  $E$  对应的最优的值, 然后同时枚举  $A, E$  贡献到  $A$  对应最优的贡献, 然后枚举  $A, B$  点对更新 dp 数组。  $O(n^2)$



# 题解

- Case 3.1: 根有一个儿子, 分别是  $x$ , 枚举  $x$  的状态  $f_{A,B}$ , 贡献到答案。
- $O(n^2)$

# 题解

- Case 3.2: 根有两个儿子, 分别是  $x, y$ , 换边顺序是  $x, y$
- 枚举  $x$  的状态  $f_{A,B}$ ,  $y$  的状态  $f_{C,E}$  它们加上  $w_A \times (d_E - d_u + 1) + w_u \times (d_B - d_u) + w_C$  会贡献到答案。
- 新增贡献分别代表:  $w_A$  上来一步然后走到  $E$ ,  $u$  往下走到  $B$ ,  $C$  上来一步。
- 发现除了  $A, E$  剩下贡献两个都是分离的, 那就先预处理每个  $A, E$  对应的最优的值, 然后同时枚举  $A, E$  更新答案。  
 $O(n^2)$

## 反思与总结

- 感觉这道题的状态，使用节点的编号比距离优越，状态数好分析，好储存，但其实并没有完全压缩好状态，这个度如何把握？
- 这种大分讨能力如何获得？在考场真的有信心调出来吗？
- 这种大分讨可以机械化吗？让程序完成。