意念力解题报告

中国人民大学附属中学董亦骐

16. Oktober 2023

1 题目内容

1.1 题目描述

给定一棵n个点的边带权的树和一个可行距离值k，称一个集合S是”合法的”,当且仅当S中任意两个不同元素在树上的距离都大于等于k。
对于$m = 1, 2, 3, ..., n$,求将这棵树的点集划分为无序的m个非空集合，即点集中每个点恰好被m个集合中的一个包含，且每个集合都是”合法的”的方案数,对998244353取模。

1.2 数据范围

$2 \leq n \leq 100000, 1 \leq k \leq 10^{14}, 1 \leq x, y \leq n, 1 \leq z \leq 10^9$。
Subtask 1(10 pts):$n \leq 10$
Subtask 2(15 pts):$n \leq 2000$,且所有边的边权均为1。
Subtask 3(15 pts):$n \leq 2000$
Subtask 4(15 pts):对于$1 \leq i \leq n-1$,第i条边满足$x = i$，$y = i + 1$,且所有边的边权均为1。
Subtask 5(15 pts):对于$1 \leq i \leq n-1$,第i条边满足$x = i$，$y = i + 1$。
Subtask 6(15 pts):所有边的边权均为1。
Subtask 7(15 pts):无特殊限制。

2 题解

2.1 Subtask 1:$n \le 10$

暴力即可。一种简单的暴力是预处理每个集合是否合法，然后直接搜索所有可能的划分并判断。
一种可能的复杂度是$O(n^2 Bell(n))$,注意到$Bell(10) = 115975$,可以通过。

2.2 Subtask 4:树是一条链，所有边权都为1

事实上，(n,k)时的答案与$(n-k+1,1)$几乎相同，唯一的区别是需要在开头多输出$k-1$个0。而(n-k+1,1)由于任何集合都合法，题面的要求恰好就是第二类斯特林数的定义。

我们将问题转化为将这棵树的每个点染上m种颜色中的一种，距离小于k的两个点不能同色，这与原问题是几乎等价的，唯一的问题是可能存在若干种颜色没有使用，也就是存在集合为空集。

为了处理这个问题，我们进行一次容斥，设m染色的答案为f_m，题目中的答案为ans_m，那么事实上,$ans_m = \frac{\sum_{i=1}^{m} \binom{m}{i} f_i (-1)^{m-i}}{m!}$,进行一些简单的转化后即可使用卷积解决，这一部分的复杂度是$O(nlogn)$

那么，对于任何子任务，只要我们能求出对于每个$1 \le m \le n$,m染色的方案数，就可以解决这一子任务。对于之后的每个子任务，我们只考虑如何计数染色方案。

对于这一子任务，求出方案数是简单的。我们考虑将链上的每个点从一端到另一端依次染色。在给第i个点染色时，这个点不能与第$max(1,i-k+1)$到$i-1$的点同色，注意到这些点显然是颜色互不相同的，也就是说给第i个点染色的方案数是$m+1-min(i,k)$，那么染色的方案数就是$\prod_{i=1}^{n}(m+1-min(i,k))$,预处理阶乘后可以使用快速幂$O(logn)$求出单个$m$的答案，总复杂度$O(nlogn)$

接下来说明为什么(n,k)时的答案与$(n-1,k-1)$是如此相似的。

事实上，可以观察到(n,k)时将树x染色的方案数就是$(n-1,k-1)$时将树$x-1$染色的方案数的x倍。进行容斥后这一点仍然成立，也就是说不存在空集的情况下，集合有序时(n,k)时划分为x个集合的方案数$(n,k-1)$划分为$x-1$个集合的方案数的x倍，集合无序时左边除以$x!$,右边除以$(x-1)!$,那么两边就恰好相等了。

因此这一子任务可以不必使用上面较麻烦的做法，直接求一行第二类斯特林数即可。

2.3 Subtask 5:树是一条链

对于这一子任务，染色方案数没有了边权为1时简单的形式。

仍然考虑从链的一端到另一端依次染色，使用双指针维护当前最靠左的不能与i同色的点，称它为l,那么显然，对于l到$i-1$间任意两点x,y,$dis(x,y)<k$,也就说l到$i-1$之间的点两两不同色，那么当前的方案数就是$m-i+l$,我们将所有这样的$i-l$组成一个数列,称它为a_1到a_n,那么对于一个m，如果它小于数列a中的最大值，显然不存在染色方案。否则，方案数为$\prod_{i=1}^{n}(m-a_i)$,这一式子可以看作是一个关于m的n次多项式，可以分治ntt求出。

而要求出$m=1,2,3,...,n$时这一多项式的值，可以看作多项式多点求值。

分治ntt和多项式多点求值的复杂度均为$O(nlog^2n)$。

2.4 Subtask 2:$n\leq 2000$,所有边权都为1

对于这一子任务，树不再是一条链，没有了链的情况下简单的染色顺序。

事实上，仍然存在一个好的染色顺序，使得我们可以简单地计算出染色方案数。

我们将点1规定为这棵树的根。

我们考虑将所有点按以1为根时的深度从小到大排序（由于对称性，可以以任何点为根)，然后依次给每个点染色。因为一个点的深度小于它的祖先的深度，也就是说排序后的序列满足任何前缀都是一个包含根的连通块。

那么，我们再给第i个点染色时，它就不能与前$i-1$个点中和它距离小于k的点同色。我们称不能与i同色的点的集合为S,下面证明，S中任意两点的距离都小于k，也就是说S中任意两点都不同色。

我们称第i个点为x,考虑S中的两个点$v1, v2$,设他们的lca为$v0$,下面我们证明$dis(v1, v2) < k$。

由前面的条件，$dep_{v1} \leq dep_x, dep_{v2} \leq dep_x, dis(x, v0) < k, dis(x, v1) < k$,考虑$dis(v1, v2) = dep_{v1} + dep_{v2} - 2dep_{v0}$,也就是说如果两个点的lca不变，而其中一个点的深度变大，那么距离也会变大。注意到如果x与$v1$不在$v0$的同一个子树里，那么$lca(x, v1) = v0$,也就是说$dis(v1, v2) \leq dis(x, v1) < k$,对$v2$同理。

而$v1$与$v2$显然不能在$v0$的同一个儿子的子树里，因为$lca(v1, v2) = v0$。所以对于每个x的可能位置,都有$dis(v1, v2) < k$,证毕。

那么既然证明了S中任意两点不同色，那么当前染色的方案数就是$m - |S|$,对于这一子任务，我们可以使用暴力的方法求出$|S|$,也就是依次枚举每个点并依次判断每个点是否属于S,预处理两点之间距离后即可$O(n^2)$解决这一子任务。

这一子任务中也不需要使用多项式技巧，可以全部用$O(n^2)$的暴力替代。

2.5 Subtask 3:$n \leq 2000$

注意到证明中并不依赖边权为1，也就是说做法是几乎相同的。
仍然使用暴力的方法求出每个$|S|$,之后的内容就和上个子任务几乎相同了。

2.6 Subtask 6:$n \leq 100000$,所有边权都为1

现在，我们的瓶颈在于快速求出每个$|S|$。考虑依次按深度从小到大排序后依次插入每个点，我们需要支持插入一个叶子，并查询当前到该叶子距离小于k的点数。

事实上，这可以用点分树实现，对于点分树上的每个点，维护两个大小为子树大小的树状数组,一个维护当前插入的点到它的距离，一个维护当前插入的点到它的父亲的距离（用于去除查询点和插入的某个点在分治中心的同一子树中的情况）即可，复杂度为$O(nlog^2n)$，事实上，这与洛谷上的点分树模板是几乎一样的。

2.7 Subtask 7:$n \leq 100000$

由于边权不为1，所以两点之间的距离可能会很大，不能简单的用树状数组维护。这一问题有若干种解决办法，如离散化，或者将树状数组替换为平衡树等。

复杂度均为$O(nlog^2n)$，可以通过本题。