T1

枚举各个质数，把是这个数倍数的点留下，跑直径

T2

先求出树的dfs序，我们可以用点对表示一种可行方案，假设制定的路径起点为a终点为b，则可以用点对(dfs\_id[a],dfs\_id[b])表示路径。 假设钥匙在节点A，宝箱在节点B，$C=LCA(A,B)$，则可以分四种情况讨论

1.C ≠ A, C ≠ B 对于这种情况，只要起点在以A为根的子树中，终点在以B为根的子树中，都可以拿到这份宝藏，而子树A中的所有节点dfs序连续，子树B同理，于是我们可以用一个矩阵表示能取得该宝藏的所有方案。

2.C = A, A ≠ B 对于这种情况，需要先求出节点D，D为路径(A,B)上最靠近A的节点，那么只要终点在子树B上，起点不在子树D上的路径，都可以拿到这份宝藏，而不在子树D上的点，可以用一个或者两个dfs序区间表示，因此可以用最多两个矩阵表示能取得该宝藏的所有方案。

3.C = B, A ≠ B 和情况2类似。

4.A = B对于这种情况，若要求出所有经过节点A的路径，矩阵数目会是n²级别的，因此反过来思考，求出所有不包含节点A的路径，对于全部这种情况来说这样矩阵数目的级别为n。可以对答案先累加宝藏权值，然后对于所有不经过该点的矩阵减去这部分权值即可。

这样我们就得到了一些带权矩阵，问题变成了在二维平面中，每次对一个矩阵内的所有点对加上一个权值，最后询问二维平面内点对的最大权值是多少，可以用扫描线+线段树解决，时间复杂度O(nlogn)

T3

高斯消元。

要求最后的乘积是完全平方数，也就是要求每个质因子都出现了偶数次，所以只跟质因子的指数的奇偶性相关。将n个数看作n个01变量，每个质数的指数和都是一个方程，高斯消元解异或方程即可。设自由元个数为k，答案即为(2^k)-1。注意到2000以内质数个数大约300个，所以复杂度为O(300^3)

整理：@SiriusRen