

8.1 CT01 题解

crazy_cloud

July 2023

1 为啥理合这

本题考察广度优先搜索序列的定义和性质。
一种可行的判断方式是：

- 序列中节点深度不下降，
- 且相邻节点若深度相同，则排在前面的节点父亲的序要比排在后面节点父亲的小。

时间复杂度 $O(n)$ 。

2 排列交换

本题考查字典序类贪心，以及图的连通性。

我们从前往后确定排列的每一位，因为是要求字典序最小，所以每一位我们都填可能的最小的数。

于是，问题变成第 i 位能填的数最小是多少。考虑枚举 i 以及后面每一位，将所有能换到 i 这个位置的数取一个最小值。

可以发现，将 A 看做图的邻接矩阵，那么能否交换其实就看两个点的连通性。这个信息预处理即可。

有一个小问题是，后面的数换到 i 这个位置需要途径 i 前面已经确定过的位置？

其实这样并不会影响正确性，我们再把这位置依次换回来即可。

时间复杂度 $O(n^2)$ 。

3 进度条

本题考察模拟的加速。

如果 $k = 1$ ，那么除法一点用都没有，答案直接是 $(n - 1) \times A$ 。

如果 $k \neq 1$ ，首先如果 n 不是 k 的倍数，那么它只能进行第一种操作，把 n 变成 $n - n \bmod k$ 再考虑。

那么现在 n 是 k 的倍数了，如果当前用一个除法比直接一步步减到 n/k 要优，那么肯定满足 $B < A \times (n - n/k)$ 。

也就是说，如果当前的 n 满足这个不等式，我们就用除法，否则用减法。
时间复杂度 $O(\log_k n)$ 。

4 跳格子

本题考查字典序类贪心。

既然要求字典序最小，那么我们肯定希望尽可能跳 a_i 步。

那如何判断我当前是否可以走 a_i 步呢？其实就是判断目的地是否存在刚好抵达 n 的方案。

我们从 n 出发反过来 DFS，把所有能走到 n 的格子都标记了。

那么贪心的时候只要跳 a_i 步的格子是被标记过的，就跳 a_i 。否则跳 b_i 。

如果走到了没被标记的格子，说明无解。如果走到了以前走过的位置，说明解无穷长。（因为我们先走严格更优的，而当时未走到答案，说明在这儿绕圈比直接走到答案更优，所以是无限长度解）

时间复杂度 $O(n)$ 。