

Assignment 4 Report

字典顺序输出整数排列问题

曹旭 大数据学院 16307110230

一、题目描述与分析

题目给定整数 $n (1 \leq n \leq 8)$ ，要求以字典顺序返回所有可能的排列。由于题目涉及到排列的各种组合，并不适合使用链表解决，因为链表的 find 函数定位的时间复杂度就有 $O(n)$ 。一开始我的想法是建一个全连接的网络，利用深度优先搜索的思路遍历与输出，但这个方法并不代码并不易于实现，所以我暂时放弃了这一策略，转为选择仅仅利用数组字典排序算法解决这一问题[1]。

二、使用数组的字典排序解决方法

使用数组的排序策略如下：首先，假设这 n 个数的某一个排列为 A_1, A_2, \dots, A_n ，然后从该序列的最右端 A_n 开始向左找出第一个比与自己相邻的右边数小的数，记其下标为 j ，即 $j = \max\{i | A_i < A_{i+1}\}$ 。然后找出 A_j 右边比 A_j 大的最小数 A_k ，交换 A_j 和 A_k ，此时序列变为了 $A_1, A_2, \dots, A_{j-1}, A_k, A_{j+1}, \dots, A_{k-1}, A_j, A_{k+1}, \dots, A_n$ 。最后将 A_{j+1}, \dots, A_n 这部分反转，即得到此序列的后一个序列 $A_1, A_2, \dots, A_{j-1}, A_k, A_n, \dots, A_{k+1}, A_j, A_{k-1}, \dots, A_{j+1}$ 。

在我的代码中，我先写好了“出 A_j 右边比 A_j 大的最小数 A_k ”，“从该序列的最右端 A_n 开始向左找出第一个比与自己相邻的右边数小的数”和“反转序列”这三个函数，在主函数中利用 while 循环输出最终结果。该方法时间复杂度为 $O(n*n!)$ ，空间复杂度为原数组大小（即 $O(n)$ ）虽然方法比较“暴力”，但是用于解决 10 以内的数字字典排序还是完全没有问题的。

以输入 $n=7$ 为例：

输出 1 2 3 4 5 6 7, 调整 $j=5, k=6$;

输出 1 2 3 4 5 7 6, 调整 $j=4, k=6$;

输出 1 2 3 4 6 5 7, 调整 $j=5, k=6$;

输出 1 2 3 4 6 7 5, 调整 $j=4, k=5$;

输出 1 2 3 4 7 5 6, 调整 $j=5, k=6$;

输出 1 2 3 4 7 6 5, 调整 $j=3, k=6$;

.....