

Web组2019秋季新人后端任务

本期任务主要是让大家了解,学习一些基本的数据结构,废话不多说了,直接冲冲冲!!!

排序~~

说到排序,学过C语言的同学肯定知道冒泡排序,但是从信息论(不知道没关系,就当我在炫技)的角度来分析,我们排好一组数据需要获得的最低信息量遍历一遍即可得到的,所以肯定有比冒泡排序更好的排序方法!!!

时间 OR 空间?

- 时间和空间是排序问题中的基本矛盾之一
- 通过学习以下算法你将学习到计算机科学家关于这个矛盾的思考
- 在学习过程中你们需要了解一些排序算法的几个基本特征: **时间复杂度,空间复杂度,稳定性**

需要你们实现的排序算法(算法名称有加粗)

- 首先还是复习一下**冒泡排序**吧
- 来一个稍微快那么一点点的**插入排序**(真的就是快那么一点点)
- 再快一点? **快速排序**(听名字就很快),听说快速排序不稳定?为啥呀?
- 听说快速排序可能会退化成冒泡排序? **堆排序**了解下?
- **归并排序**表示:我也很不错!!
- 我编不下去了后面还有一些算法也要实现:**计数排序,桶排序**

数据结构

栈,队列

- 用两个栈实现一个队列
- 实现一个优先队列
- 使用栈实现中缀表达式转后缀表达式的算法(计算的数据均为十进制正整数,运算符为加减乘除)

二叉树

- 掌握二叉树的 DFS、BFS 遍历(允许使用编程语言自带的变长数组,队列之类的数据结构)
 - DFS前序遍历 (先根周游)
 - DFS中序遍历 (中根周游)
 - DFS后序遍历 (后根周游)
 - 以上三种均要用递归,非递归分别实现一遍
 - BFS层级优先遍历非递归实现
- 掌握一种特殊的二叉树——二叉搜索树
 - 实现一颗二叉搜索树,并包含以下操作

- insert 插入元素
- find 搜索元素
- delete 删除元素
- 学习一种特殊的二叉搜索树——AVL 树
 - 只用看一下基本概念、不需要实现
 - 理解 AVL 树比普通的二叉搜索树好在哪里、解决了什么痛点、有哪些不足

哈希表

- 只用看一下基本概念、不需要实现
- 理解常见概念，如哈希冲突、哈希函数等等

并查集

- 允许使用编程语言自带的变长数组,队列之类的数据结构
- 理解基本概念，实现一个并查集
- 实现并查集的操作
 - merge求并集
 - findRoot求一个元素的父节点
- 实现并查集的优化
 - 按秩合并
 - 路径压缩

基础的图算法

- 允许使用编程语言自带的变长数组,队列之类的数据结构
- 学习 图的邻接矩阵表示法
- 学习并掌握 基础遍历算法
 - DFS
 - 递归实现
 - 非递归实现
 - 基于 DFS 的拓扑排序
 - BFS
 - 非递归实现
- 最小生成树算法
 - 基于之前实现的并查集，实现 Kruskal 算法
 - 基于二叉堆的优先队列，实现 Prim 算法
 - 允许使用priority_queue或之前自己实现的优先队列
- 单源最短路径算法
 - 基于二叉堆的优先队列，实现 Dijkstra 算法
 - 比较和 Prim 算法思路、实现方法上的异同点
 - 允许使用priority_queue或之前自己实现的优先队列

要求

- 语言不限
- 除非明确说明,不准使用语言自带的标准库自带的数据结构和相关算法
- 任务时间 2019/10/28-2019/11/2

参考

- 《算法导论》
- 《数据结构与算法：C 语言描述》