重庆大学本科学生实验项目任务书

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 实验题目 | 排序算法实践 | | | |
| 实验时间 | 2022.10.20 | 实验地点 | | 宿舍 |
| 实验性质 | □验证性 √设计性 □综合性 | | | |
| 实验目的  1. 掌握各类排序算法的基本原理及时空复杂度分析方法  2. 训练使用排序算法，通过编程解决不同难度问题的实践能力 | | | | |
| 实验内容   1. 排序算法基本原理及时空复杂度分析方法 2. 在线编程解决3道不同难度的编程题，共40分   3． 在线编程时间（答题时长）150分钟，从开始答题时刻起算  4． 在线程序测评平台采用PTA (https://pintia.cn/)  5． 每人必须独立完成编程，可查阅教科书、PPT等资料，不得相互抄袭以及抄袭网上已有的程序  6． 实验课结束后，会对所有程序进行查重，如检测出有抄袭的程序，成绩计零分处理  注意：最后提交完整的实验报告，包括对每道题的算法思路、代码描述、复杂度分析等内容。 | | | | |
| 参考资料：   * Data Structures and Algorithm Analysis (C++ Version) Clifford A. Shaffer   + Introduction to Algorithms, 3rd Edition, MIT Press, T.H. Cormen, et al.   + 《数据结构（ C 语言版）》，严蔚敏，吴伟民编著，清华大学出版社 | | | | |
| 任务下达日期 2022 年 10月 10 日 | | | 完成日期 2022 年10 月 23 日 | |

说明：学院、专业、年级均填全称，如：计算机学院、计算机科学与技术、2018。

**《数据结构与算法》实验报告**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **年级、专业、班级** | | 计算机学院、计算机科学与技术、卓越一班 | | | **姓名** | **罗帆靖** |
| **实验题目** | 排序算法实践 | | | | | |
| **实验时间** | **2022.10.20** | | **实验地点** | **寝室** | | |
| **实验成绩** |  | | **实验性质** | **□验证性** √**设计性 □综合性** | | |
| 教师评价：  □算法/实验过程正确； □源程序/实验内容提交 □程序结构/实验步骤合理；  □实验结果正确； □语法、语义正确； □报告规范；  其他：  评价教师签名： | | | | | | |
| 一、实验目的  1. 掌握各类排序算法的基本原理及时空复杂度分析方法  2. 训练使用排序算法，通过编程解决不同难度问题的实践能力 | | | | | | |
| 二、实验项目内容    实验课题1：  **题目内容**：选择排序  **解题思路**：标准的选择排序题，因为涉及到循秩访问，所以采用以数组形式保存的线性表存储数据。  程序每次进行一次外层循环即可确定一位的数字，第i次外层循环开始时（此时从0到i-1的数位已排序），先假定第i位的数字是最小的，再继续向后索引对比（内层循环），一旦遇到比当前最小值还要小的数字，就替换当前最小值并且记录下标。等内层循环结束的时候，若第i位的数字不是最小值，即当前最小值不在第i位，则进行交换，将当前最小值放在第i位，至此0到i位的数位已排序。  每完成一位数的排序即输出一遍排序过程。  **源代码：**      **时间与空间复杂度分析：**  **时间：**因未对选择排序优化，因此无论最好最坏情况下，内层循环为n-i，外层循环为n-1，其他操作均为常数级，因此总体时间复杂度为O(n^2)；  **空间：**因为排序过程中仅用到了数组本身的空间以及常数级的临时变量，因此空间复杂度为O(1)  实验课题2：  **题目内容:** 直接插入排序  **解题思路：**标准的插入排序题。每一轮循环将前面的i位数据视为有序数据，在此轮循环中，向前逐一对比需插入的数与各位数的大小，找到合适的位置并插入，并在过程中将数据向右移位以腾出位置以便数据插入。每轮循环中前i位数据的有序性得以维护。  **源代码：**    **时间与空间复杂度分析：**  **时间：**最优情况下(非降序)，每次内层循环比较一次，则O(N);最坏情况下（非升序），每次内层循环比较i次，则O(N^2);平均情况下，为O(N^2);  **空间:** 引入了常数个辅助变量，与问题规模无关，因此为O(1);  实验课题3：  **题目内容:** 贝茜放慢脚步  **解题思路:**  首先审题，将应用题转化为算法题，题目中有两个减速因素，一个是到达一定的时间，一个是到达一定的路程，且两者之间相互联系，先后到达的顺序对题目影响；我们可以模拟走的过程，利用时间为主要变量去推算哪个影响因子会来的更早。因此我们引入S，T，V三个变量分别记录当前的三个状态。将时间减速因素和路程减速因素分别放置在两个数组中（分别为a,b），并已经单独做好排序，假设当前面临两个减速因素，ai和bj，那么到达ai所需的时间为ai-t，到达bj所需要的时间为(bj-s)/v,比较两者谁先来临即可，同时更新当前S，T，V三个参数。  **源代码：**        **时间与空间复杂度分析：**  **时间：**读数据所需O(N),algorithm库所提供的sort函数使用类似快排的操作，因此对a,b数组排序所需时间复杂度为O(n\*log2(n)),主算法部分采用类似二分归并排序算法，时间复杂度为O(N);综上此算法时间复杂度为O(n\*log2(n))；  **空间：**引入了两个临时vector储存减速因子，其余引入常数级辅助变量，因此空间复杂度为O(N); | | | | | | |
| 三、思考题    用单链表存放一组数据并进行排序，插入排序、冒泡排序、归并排序，快速排序中，哪些算法能实现与顺序表相同的时间复杂度？简要说明排序过程。要求排序过程中，除输入的单链表外，算法的空间复杂度为O(1)。  由于没有对数组的中间项进行随机访问，各类算法都是采取逐项访问的方式，因此插入排序、冒泡排序、归并排序，快速排序都可以实现与顺序表相同的时间复杂度；但插入排序、冒泡排序满足空间复杂度，归并不满足空间复杂度,倘若递归开栈不算入空间复杂度，则快排也满足。  插入排序：令数组后半部分为有序部分，每次取出一元素向后遍历，直到找到合适位置，断链插入。时间复杂度O(N^2),空间复杂度O(1);  冒泡排序：对每一对相邻元素比较，从开始第一对到最后一对，将较大的数放在右边（每次的时间复杂度为常数级），一次循环过后，最后一个元素就是最大的元素。时间复杂度为O(N^2) ,空间复杂度O(1);  归并排序：采用分治法的方式，将元素递归分成n/2，直到得到n个长度为1的数列（相当于有序数列）；将已排序的数列两两排序合并，重复以得到长度为n的有序数列。因为排序需要临时申请一个长度为n的数组来存放结果，所以空间复杂度为O(n)。时间复杂度为O(nlogn);  快速排序：选择一个元素为基准轴，因为是单链表所以与普通快排不相同，此时两个指针都从左向右走，r指针在l指针右1，若逆序就交换，两者都向右走；若不逆序就r走l不走。用此方法可以找到基准轴的位置，并且实现基准轴左边都比其小，右边都比其大。递归则可实现。时间复杂度为O(nlogn),时间复杂度为O(1)（若不算入递归开栈）; | | | | | | |