

**算法设计与分析**

**实 验 报 告**

学 号： 2201110126

学生姓名： 宋致远

班 级： 2011101

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **评价** | **优** | **良** | **中** | **差** | **评语：** |
| **求解思路** |  |  |  |  |
| **伪代码** |  |  |  |  |
| **算法分析** |  |  |  |  |
| **测试结果** |  |  |  |  |
| **报告格式** |  |  |  |  | **得分：** |

**2022-04**

**算法设计与分析 实验报告**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 姓名 | 宋致远 | | 院系 | 软件学院 | | 学号 | 2201110126 |
| 任课教师 | | 王金宝 | | | 指导教师 | 王金宝 | |
| 实验地点 | | 研究院 | | | 实验时间 | 2022年 4月 4日 | |
| 实验名称 | | 使用分治技术的算法设计与实现 | | | | | |
| 同 组 人 | | 无 | | | | | |
| 实验内容背景 | | | | | | | |
| 逆序数是数组中元素有序性程度的度量，用于衡量违背排序的元素对的数量，直观的方法比较数组中任意一对元素可以计算逆序数，代价为O(n2). 本次实验，请大家设计分治算法，更高效地求解逆序数。使用高级程序语言实现设计的算法，并与直观算法进行运行时间比较。 | | | | | | | |
| 实验内容 | | | | | | | |
| **1、问题**  设*A*[1:*n*]是由不同实数组成的数组，如果 *i*<*j*且*A*[*i*]>*A*[*j*]，则称实数对 (*A*[*i*], *A*[*j*])是该数组的一个反序。如， 给定*A*=[3,5,2,4]，则该数组存在 3个反序(3,2)、(5,2)和(5,4)。反序的个数可以用来衡量一个数组的无序程度。设计一个分治算法（要求时间复杂度严格低于*n*2），计算输入数组*A*[1:*n*]中反序的个数。生成大小为10、102、103、104、105的数组，其中包含大小随机的元素，每个大小的数组生成100次，并在不同大小的数组上比较分治算法和直观算法的平均运行时间。（用两条折现图形化展示比较结果，横轴为数组大小，纵轴为两个算法的平均运行时间。）  **2、问题求解思路**（阐述分治算法中分、治、和的思路）  2.1、Divide：把问题数组分为两份，每一份用分治算法的思想再分下去直到分的数组只有2个元素  2.2、Conquer：当分成2份数组时，每个数组头有一指针，比较a数组指针和b数组指针指向值的大小，小的则被储存在临时数组里，并指针后移，由此让所有数组进行一轮排序。其中当b数组指针指向的值小于a的指针的时候，则证明出现了逆序数，出现的组数为a数组还没有被遍历的元素的数量  2.3、Merge： 临时数组排序完成后，递归的合并回原数组，逆序数的组数就是之前的所有组数加起来  **3、算法伪代码**（非程序代码）  Sort（数组）{  If（数组元素>2）{  sort（前半数组）  sort（后半数组）  merge（前半数组，后半数组）  }  }  Merge（前半数组，后半数组）{  for(k=0,i=0,j=0;i<前半数组长度&&j<后半数组长度;k++) {  if(前半数组[i]< 后半数组[j]) {  临时数组[k] = 前半数组[i++];  }else {  //统计逆序对个数  逆序对个数 += 前半数组未遍历的元素个数  临时数组[k] =后半数组[j++];  }  }  if(任意数组遍历完了另一个还没) {  另一个数组剩下的填进临时数组  }  }  **4、算法复杂性分析**  T（n）=2T（n/2）+o(n)  T（n）=o（nlogn）  **5、测试结果**（说明程序中使用的主要数据结构及其上的操作）  数组 进行了读写  Input：(1,2,6,3,4,7,5)  Output: 3  暴力算法的时间代价与分治算法的时间代价对比，当数组规模越来越大时，两种算法的代价产生了明显的差距。  IMG_256 | | | | | | | |
| 实验结论（结果分析、遇到的困难和解决方法等） | | | | | | 备注 |  |
| 遇到了无法一次性完成所有问题的困难，采取分治，将大问题分成小问题递归的解决  通过本次实验，掌握了分治算法的使用和分析，对算法的推理和使用有了更加深刻的理解，为以后将该算法活用在项目中打下了基础。 | | | | | | | |