

**算法设计与分析**

**实 验 报 告**

学 号： 2201110228

学生姓名： 杨志超

班 级： 2011102

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **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 | 杨志超 | | 院系 | 软件学院 | | 学号 | 2201110228 |
| 任课教师 | | 王金宝 | | | 指导教师 | 王金宝 | |
| 实验地点 | | 线上会议室 | | | 实验时间 | 年 月 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。  **2、问题求解思路**  2.1、Divide：  对于一个列表，将其分为数量尽量相同的两部分，在两部分找出逆序对，并在过程中进行排序。  2.2、Conquer：  对于一个给定的列表，如果其长度为1，返回自身和0，（其中0是逆序对的个数），否则继续进行分割。  2.3、Merge：  根据分完后的两半的结果，使用比较的方式计算两半之间的逆序对个数（也即逆序对中小的在前面一半，逆序对的大的在后面一半的逆序对），同时在过程中进行排序，最后将排序出的列表和逆序对个数与两半分别的逆序对个数的加和返回。  **3、算法伪代码**  FindVersePairNum([]int A):  If(len(A)==1)  return A,0;  else:  A1 = A[0:len(A)/2];  A2 = A[len(A)/2,len(A)];  A1,num1 = FindVersePairNum(A1);  A2,num2 = FindVersePairNum(A2);  Empty A;  ans = 0  j = 0  For i = 0 to len(A1)-1:  While A2[j]<=A1[i] and j<len(A2):  j++;  A.append(A2[j]);  A.append(A1[i]);  If j==len(A2):  continue;  ans+=len(A1)-j:  return A, ans+num1,num2  **4、算法复杂性分析**  对于长度为1的列表，O(1)时间可以返回结果，对于子问题已经解决的问题，Merge操作在O(n)时间内可以完成，根据master定理，总的时间复杂度为O(nlogn)。  **5、测试结果**  1. 使用的数据结构:列表，能够进行截取、清空、按索引访问、尾部添加等操作  2. 直观算法(O(n2))与分治算法(O(nlogn))在不同测试集上运行速度的比较: | | | | | | | |
| 实验结论（结果分析、遇到的困难和解决方法等） | | | | | | 备注 |  |
|  | | | | | | | |