**《数据结构综合设计》实验报告**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 学院 | 人工智能与大数据学院 | 专业 | 虚拟现实技术 | 班级 | 21级1班 | 学生姓名 | 王涵 |
| 实验  周次 | 13-14 | 实验  日期 | 2023.5.29 | 学时 | 6 | 教师姓名 | 李昊康 |
| 项目名称 | | 排序的应用 | | | | | |
| 实验  类别 | 🗹验证型实验 🞎设计型实验 🞎综合型实验 🞎其它 | | | | | 成绩：85 | |
| 1. 实验目的及具体要求   实验目的：  1.实现多种类型的排序算法（插入排序、交换排序、选择排序、归并排序等）；  2.理解排序过程；  3.计算比较次数和移动次数，对比分析算法性能的优劣与适用场景；  具体要求：  编写程序实现插入排序、希尔排序、冒泡排序、快速排序、简单选择排序、堆排序。中任意3种。   1. 实验仪器、设备和材料   硬设备：PC机  软件环境：Windows VS2019   1. 实验内容、步骤及实验数据记录   插入排序：  #include<stdio.h>  #include<string.h>  #define len 5  void insertSort(int a[])  {  int i,j,temp;  for(i=1;i<len;i++)  {  temp = a[i];  //当前数小于前一位数时  if(a[i] < a[i-1])  {  //将子序列重新排列为有序序列  for(j=i-1;temp<a[j];j--)  {  a[j+1] = a[j];  }  a[j+1] = temp;  }  }  }  int main()  {  int a[] = {45,32,56,71,12};  int i;  printf("未排序前：\n");  for(i=0;i<len;i++)  {  printf("%d ", a[i]);  }  printf("\n经过直接插入排序后：\n");  insertSort(a);  for(i=0;i<len;i++)  {  printf("%d ", a[i]);  }  }    希尔排序：  #include<stdio.h>  #include<string.h>  #define len 5  void shellSort(int a[])  {  int i,j,dk,temp;  //增量dk的变化，dk = dk/2  for(dk=len/2;dk>0;dk/=2)  {  for(i=dk;i<len;i++)  {  //需将a[i]插入有序增量子表中  if(a[i] < a[i-dk])  {  //存到临时变量中  temp = a[i];  for(j=i-dk;j>=0&&temp<a[j];j-=dk)  {  //统一记录后移，查找插入的位置  a[j+dk] = a[j];  }  a[j+dk] = temp;  }  }  }  }  int main()  {  int a[] = {45,32,56,71,12};  int i;  printf("未排序前：\n");  for(i=0;i<len;i++)  {  printf("%d ", a[i]);  }  printf("\n经过直接插入排序后：\n");  shellSort(a);  for(i=0;i<len;i++)  {  printf("%d ", a[i]);  }  }  冒泡排序：  #include<stdio.h>  int main(void)  {  int a[10]={6,4,3,2,7,8,9,10,1,5};  int i,k,w;  for(i=0;i<9;i++)  {  for(k=0;k<9-i;k++)  {  if(a[k]>a[k+1])  {  w=a[k];  a[k]=a[k+1];  a[k+1]=w;  }  }  }  for(i=0;i<10;i++)  {  printf("%d ",a[i]);  }  }  分析：  1.实现多种类型的排序算法（插入排序、交换排序、选择排序、归并排序等）；  2.理解排序过程；  3.计算比较次数和移动次数，对比分析算法性能的优劣与适用场景； | | | | | | | |

说明：1. 实验周次：填写实际上课周，如第5-8周上课填“5-8”或第10周上课填“10”。

1. 实验报告各部分内容需详实填写，按实验指导书上的评分标准给出分数。
2. 实验类型参考实验类型说明文件。