**《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3班 | 学生姓名 | 罗美惠 |
| 实验  周次 | 13-14 | 实验  日期 | 2023.5.25 | 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：93 | |
| 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 2. 插入排序：   #include <stdio.h>  void insertionSort(int arr[], int n) {  int i, key, j;  //遍历数组  for (i = 1; i < n; i++)  {  //将当前元素存储到key中，为其在后续操作中腾出位置  key = arr[i];  j = i - 1;  //移动比key大的元素到当前位置之后  while (j >= 0 && arr[j] > key)  {  arr[j + 1] = arr[j];  j = j - 1;  }  //将key插入到正确的位置  arr[j + 1] = key;  }  }  int main() {  int arr[] = { 12, 11, 13, 5, 6 };  int n = sizeof(arr) / sizeof(arr[0]);  int i;  printf("Original Array: ");  for (i = 0; i < n; i++)  printf("%d ", arr[i]);  insertionSort(arr, n);  printf("\nSorted Array: ");  for (i = 0; i < n; i++)  printf("%d ", arr[i]);  return 0;  }   1. 希尔排序   #include <stdio.h>  void shellSort(int arr[], int n) {  int i, j, gap, temp;  //循环选取增量gap，每次gap减半  for (gap = n / 2; gap > 0; gap /= 2) {  //在每个"分组"内进行插入排序  for (i = gap; i < n; i++) {  temp = arr[i];  j = i;  //移动比temp大的元素到当前位置之后  while (j >= gap && arr[j - gap] > temp) {  arr[j] = arr[j - gap];  j -= gap;  }  //将temp插入到正确的位置  arr[j] = temp;  }  }  }  int main() {  int arr[] = { 12, 11, 13, 5, 6 };  int n = sizeof(arr) / sizeof(arr[0]);  int i;  printf("Original Array: ");  for (i = 0; i < n; i++)  printf("%d ", arr[i]);  shellSort(arr, n);  printf("\nSorted Array: ");  for (i = 0; i < n; i++)  printf("%d ", arr[i]);  return 0;  }   1. 冒泡排序   #include <stdio.h>  void bubbleSort(int arr[], int n) {  int i, j, temp;  for (i = 0; i < n - 1; i++) {  //从每个元素开始进行比较，将大的元素冒泡到顶端  for (j = 0; j < n - i - 1; j++) {  if (arr[j] > arr[j + 1]) {  temp = arr[j];  arr[j] = arr[j + 1];  arr[j + 1] = temp;  }  }  }  }  int main() {  int arr[] = { 12, 11, 13, 5, 6 };  int n = sizeof(arr) / sizeof(arr[0]);  int i;  printf("Original Array: ");  for (i = 0; i < n; i++)  printf("%d ", arr[i]);  bubbleSort(arr, n);  printf("\nSorted Array: ");  for (i = 0; i < n; i++)  printf("%d ", arr[i]);  return 0;  }  分析：  1.插入排序：  IMG_256  IMG_256  2.希尔排序  IMG_256  IMG_256   1. 冒泡排序   IMG_256IMG_256 | | | | | | | |

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。