软件学院《数据结构》课程实验报告

 《实验内容不可缺少，格式不重要，不要修改蓝色文字，会影响预审，Ver1》

《实验报告文件名格式：200020002000-李同学-实验2-排序算法.docx》

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 班级 | 21.3 | 学号 | 202100300157 | 姓名 | 孙荣骏 |
| 【实验题目、实验目的、实验内容:(从大纲复制到下面框中，要文字不要截图)】 | | | | | |
| 实验2 排序算法  1. 实验目的   掌握三种排序方法冒泡、插入、归并的实现思想。   1. 实验内容 2. 最多接受18个不为零的正整数进行排序，如果中间输入0则代表提前结束输入，0之前输入几个数就用几个数参与排序，0不参与排序。 3. 数字选择排序方法，1-Bubble Sort,2-Insert Sort,3-Merge Sort(注意大小写要区分)。 4. 使用所选排序方法的排序，结果输出所用方法以及结果，每个数之间用“，”隔开，中间不要有空格。 5. 输入输出请严格按下面要求的格式实现，不能少任何一行文字。 | | | | | |
| 【编译工具及其版本:(例如：Visual C++、My Eclipse、Turbo C、gcc、CLion、Dev-C++、Visual Studio、Xcode、Vim、Qt Creator等。内容放在下面框中，要文字不要截图)】 | | | | | |
| Dev-C++ | | | | | |
| 【结论分析、实验体会:(如何找到错误原因、如何排除错误、如何找到解决方法、以后如何防范类似错误)】 | | | | | |
| 实验时本来用的Bubble Sort，Insert Sort，Merge Sort当方法名，报错后才想起来方法名不能有空格  本来在main方法中将scanf、cin、cout方法混用了，结果出现了一些错误，查阅后知道这些方法最好不要一起用，因为底层逻辑不一样，会引起一些错误，所以改成了只用cin和cout  循环读入输入时，因为语句顺序写的有问题，导致我写的n==18但实际可以接收19个数才自动停止，通过改正可以正确的接收18个数然后自动停止 | | | | | |
| 【实验全部代码:(内容放在下面框中，要文字不要截图)】 | | | | | |
| #include <bits/stdc++.h>  #include<cstring>  using namespace std;  void BubbleSort(int arr[], int len)  {  int i, j; int temp;  for (i = 0; i < len - 1; i++)  for (j = 0; j < len - 1 - i; j++)  if (arr[j] > arr[j + 1])  {  temp = arr[j];  arr[j] = arr[j + 1];  arr[j + 1] = temp;  }  }  void InsertSort(int arr[],int len)  {  int tmp;  int i;  int j;  for (i = 1;i < len;i++)  {  tmp = arr[i];  for (j = i - 1;j >= 0 && arr[j] > tmp;j--)  {  arr[j + 1] = arr[j];  }  arr[j + 1] = tmp;  }  }  void method(int s[],int n){  for(int l=0;l<n-1;l++)cout<<s[l]<<",";  cout<<s[n-1]<<endl;  return;  }  void MergeSortArr(int arr[],int left,int mid,int right){  int i = left,j = mid + 1;  int k=0;  int temp[right-left+1];  while(i <=mid && j <= right){  if(arr[i] < arr[j])  temp[k++] = arr[i++];  else  temp[k++] = arr[j++];  }  while(i <= mid){  temp[k++] = arr[i++];  }  while(j <= right){  temp[k++] = arr[j++];  }  for(i = 0;i < k;i++)  arr[left++] = temp[i];  }  void MergeSort(int arr[],int n){  int i,j;  int left,mid,right;  for(i=2; ;i \*= 2){  for(j = 0;j < n;j += i){  left = j;  mid = (j + j + i - 1) / 2 < n? (j + j + i - 1) / 2 : n-1;  right = (j + i -1) < n? (j + i -1) : n-1;  MergeSortArr(arr, left, mid, right);  }  if(i > n)  break;  }  }  int main(){  cout<<"Input"<<endl;  int s[20];int n=0;  int k=-1;  while(1){  if(n==18)break;  cin>>k;  if(k==0)break;  s[n++]=k;  }  cout<<"1-Bubble Sort,2-Insert Sort,3-Merge Sort"<<endl;  int a=0;  cin>>a;  cout<<"Output"<<endl;  if(a==1){  cout<<"Bubble Sort"<<endl;  BubbleSort(s,n);    }else if(a==2){  cout<<"Insert Sort"<<endl;  InsertSort(s,n);    }else if(a==3){  cout<<"Merge Sort"<<endl;  MergeSort(s,n);    }    method(s,n);  cout<<"End";  return 0;  } | | | | | |