**Southwest University**

**School of Computer & Information Science**

Experiment Report

**Course：**  C Programming

**Topic ：**  Function and Array

**Grade&class：**Grade 2021  **Class CS3**

**Student Name：**  蔡前哲

**Submission Date：**  2022年4月30日

|  |
| --- |
| 1. ****Description of the problem （main task and the problem you resolved）****   Write a program to sort the ten integers in ascending order by Bubble sort algorithm or Exchange Maximum sort (select) algorithm, use the Binary search algorithm to search the number(recursive), and then if the data doesn’t exist, insert it into the array (After inserting the data, the array is still ordered).  The input specifications are these:   1. Read the ten integers from the file (The ten integers in the file are produced randomly, using rand () or srand () function\*.). 2. Read the searching data form the keyboard.   The output specification is this:  The dates before sorting, after sorting and inserting;  The user interface specification is this:  Use Menu to choose reading the dates, sorting algorithms and searching the data (while +switch);  The 1st level Menu is this:  ---------------------------------------------   1. Reading the data from the file 2. Sorting algorithms 3. Binary search algorithm   When choose 2, the 2nd level is this:  ---------------------------------------------   * 1. Bubble sort algorithm   2. Exchange Maximum sort (select) Algorithm   **I should achieve the above functions. Emphasis is on hierarchical menus and interactive interfaces. Implement corresponding functions according to user selection** |
| 1. ****Basic Conceptions and Principles****   ****1. File operations****  **I need to implement reading in and outputting data from a file. Using file pointers and the fscanf() and fprintf() functions.**  ****2. Pointer variable****  **I need to swap variable values using pointers to achieve sorting.**  ****3.Bubble Sort****  **Bubble Sort is a relatively simple sorting algorithm in the field of computer science.**  **It repeatedly walks through the column of elements to be sorted, compares two adjacent elements in turn, and swaps them if the order is wrong (e.g., big to small, first letter from Z to A). The work of visiting elements is repeated until no adjacent elements need to be exchanged, that is, the element column has been sorted.**  **The name of this algorithm comes from the fact that the smaller elements will slowly "float" to the top of the sequence (in ascending or descending order) through exchange, just as the bubbles of carbon dioxide in carbonated drinks will eventually float to the top, hence the name "bubbling" Sort".**  ****4.Select Sort****  **Selection sort is a simple and intuitive sorting algorithm. Its working principle is: first select the smallest (or largest) element from the data elements to be sorted, store it at the beginning of the sequence, and then find the smallest (largest) element from the remaining unsorted elements, and then placed at the end of the sorted sequence. And so on, until the number of all data elements to be sorted is zero. Selection sort is an unstable sorting method.** |
| 1. ****Design Scheme（your plan, includes the structure of your program and the necessary description）****   ****The function list:****  **void CreatDataToFile(); //随机生成n个数**  **void Read(); //从文件读数据**  **void FirstMenu(int flag); //一级菜单 flag 0第一次输入 1输入错误后重新输入**  **void swap(int \*a, int \*b); //交换两个数的值**  **void BubbleSort(); //冒泡排序**  **void SelectSort(); //选择排序**  **void SecondMenu(int flag); //二级菜单**  **void Search(); //搜索插入**  **int BinarySearch(int num); //二分查找**  ****The structure chart of program is:****  ****The data flow diagram for the general program is****  **The data source is selected by the user. Automatically generated by the program or manually entered by the user.** |
| ****4、Resource Code（Detailed comments are required）****  **#include<stdio.h>**  **#include<stdlib.h>**  **#include<time.h>**  **#include<math.h>**  **#define FOR(i,a,b) for(int i = a; i <= b; i++)**  **const double eps = 1e-6; //浮点数精度**  **const int n = 10; //数据范围**  **int dt[100], dts[100], dti[100]; //原数据与改变后的数据**  **int quit = 0, Runtime = 0; //退出标志 第一次运行标记**  **//函数声明部分**  **void CreatDataToFile(); //随机生成n个数**  **void Read(); //从文件读数据**  **void FirstMenu(int flag); //一级菜单 flag 0第一次输入 1输入错误后重新输入**  **void swap(int \*a, int \*b); //交换两个数的值**  **void BubbleSort(); //冒泡排序**  **void SelectSort(); //选择排序**  **void SecondMenu(int flag); //二级菜单**  **void Search(); //搜索插入**  **int BinarySearch(int num); //二分查找**  **//主程序**  **int main()**  **{**  **FirstMenu(0);  //一级菜单**  **return 0;**  **}**  **//随机生成n个数**  **void CreatDataToFile()**  **{**  **char slct; //读入用户选择**  **int f = 0, dataMin, dataMax;  //标记是否需要指定范围**  **//-----------------是否生成数据---------------**  **printf("Is there a need to automatically generate %d data?(Y/N)\nIf you choose NO, you will need to manually enter the data into \"data.txt\" yourself.\nPlease enter...", n);**  **scanf("%c", &slct);**  **getchar(); //清除缓冲区**  **if(slct == 'N' || slct == 'n') return;**  **//-----------------是否指定范围----------------**  **printf("\nDo you need to specify a data range?\nIf so, please enter 'Y' followed by two integers as the range, otherwise please enter 'N'.\nPlease enter...");**  **scanf("%c", &slct);**  **getchar();**  **if(slct == 'Y' || slct == 'y')**  **{**  **f = 1;**  **scanf("%d%d", &dataMin, &dataMax);**  **}**  **//-------------------随机生成数值----------------**  **FILE \*fp;**  **fp = fopen("data.txt","w");**  **srand(time(0)); //将时间作为随机数种子**  **FOR(i,1,n)**  **{**  **if(f) fprintf(fp,"%d ", rand()%(dataMax-dataMin+1)+dataMin);**  **else fprintf(fp,"%d ", rand());**  **}**  **fclose(fp);**  **printf("Data has been generated into \"data.txt\".\n");**  **//system("pause");**  **//system("cls"); //清屏**  **}**  **//从文件读数据**  **void Read()**  **{**  **FILE \*fp;**  **fp = fopen("data.txt","r");**  **FOR(i,1,n)**  **fscanf(fp,"%d",&dt[i]);**  **fclose(fp);**  **printf("Data has been read in from \"data.txt\".\n\n");**  **}**  **//交换两个数的值**  **void swap(int \*a, int \*b)**  **{**  **int t = \*a;**  **\*a = \*b;**  **\*b = t;**  **}**  **//冒泡排序**  **void BubbleSort()**  **{**  **int swapflag; //是否发生交换**  **FOR(i,1,n)  //复制数据**  **dts[i] = dt[i];**  **FOR(i,1,9)**  **{**  **swapflag = 0;**  **FOR(j,1,n-i)**  **if(dts[j] > dts[j+1])**  **{**  **swapflag = 1;**  **swap(&dts[j], &dts[j+1]);**  **}**  **if(!swapflag) break; //排序完成 退出**  **}**  **printf("Bubble sort completed.\n\n");**  **//输出到屏幕**  **printf("    Raw data:\n    ");**  **FOR(i,1,n)**  **printf("%d ",dt[i]);**  **printf("\n    Sorted data:\n    ");**  **FOR(i,1,n)**  **printf("%d ",dts[i]);**  **printf("\n\n");**  **FirstMenu(0);**  **}**  **//选择排序**  **void SelectSort()**  **{**  **FOR(i,1,n)  //复制数据**  **dts[i] = dt[i];**    **for(int i = n; i >= 1; i--)**  **{**  **int maxx = i;**  **FOR(j,1,i-1)**  **if(dts[j] > dts[maxx]) maxx = j;**  **if(maxx != i)**  **swap(&dts[maxx], &dts[i]);**  **}**  **printf("Exchange Maximum sort (select) completed.\n\n");**  **//输出到屏幕**  **printf("    Raw data:\n    ");**  **FOR(i,1,n)**  **printf("%d ",dt[i]);**  **printf("\n    Sorted data:\n    ");**  **FOR(i,1,n)**  **printf("%d ",dts[i]);**  **printf("\n\n");**  **FirstMenu(0);**  **}**  **//二级菜单**  **void SecondMenu(int flag)**  **{**  **float fun;**  **if(!flag) printf("  Please select a sort algorithm.\n    2.1 Bubble sort algorithm.\n    2.2 Exchange Maximum sort (select) Algorithm.\n  Please enter...");**  **scanf("%f", &fun);**  **if(fabs(fun-2.1) < eps) BubbleSort();**  **else if(fabs(fun-2.2) < eps) SelectSort();**  **else**  **{**  **printf("Invalid input, please enter a number again...");**  **SecondMenu(1);**  **}**  **}**  **//二分查找插入**  **int BinarySearch(int x)**  **{**  **FOR(i,1,n)**  **dti[i] = dts[i];**  **int l = 1, r = n;**  **while(l < r)**  **{**  **int mid = (l+r)>>1;**  **if(dti[mid] >= x) r = mid;**  **else l = mid+1;**  **}**    **if(dti[l] == x) printf("%d has been found, at the %d position in Sorted Data.\n\n",x,l);**  **else //未找到 插入数字**  **{**  **int tmp = dti[l];**  **dti[l] = x;**  **FOR(i,l+1,n+1)**  **swap(&tmp, &dti[i]);**  **printf("%d has been inserted into the %d position of the sorted data.\n\n",x,l);**  **//输出到屏幕**  **printf("    Sorted data:\n    ");**  **FOR(i,1,n)**  **printf("%d ",dts[i]);**  **printf("\n    Inserted data:\n    ");**  **FOR(i,1,n+1)**  **printf("%d ",dti[i]);**  **printf("\n\n");**  **}**  **}**  **//搜索插入**  **void Search()**  **{**  **int num;**  **printf("Please enter the number you want to find, if it does not exist, it will be inserted.\n  Please enter...");**  **scanf("%d", &num);**  **BinarySearch(num);**  **}**  **//一级菜单 flag 0第一次输入 1输入错误后重新输入**  **void FirstMenu(int flag)**  **{**  **if(quit) return;**  **int fun;**  **if(!flag) printf("Please select a function to execute.\n  1.Reading the data from the file.\n  2.Sorting algorithms.\n  3.Binary search algorithm.\n  4.Exit this program.\nPlease enter...");**  **scanf("%d",&fun);**  **getchar(); //清除缓冲区**  **switch(fun)**  **{**  **case 1:**  **CreatDataToFile(); //随机生成n个数到文件中**  **Read();**  **FirstMenu(0);**  **break;**  **case 2:**  **SecondMenu(0);**  **FirstMenu(0);**  **break;**  **case 3:**  **Search();**  **FirstMenu(0);**  **break;**  **case 4:**  **quit = 1;**  **return;**  **default:   //输入无效**  **printf("Invalid input, please enter a number again...");**  **FirstMenu(1);**  **}**  **}** |
| 1. ****Instructions (Give the steps and necessary screenshot to show how to run your program)****   Run exe program.  **Enter number 1.**  **Please enter the corresponding letters according to the program prompts and your own needs.** **I typed y.**  I need data from 1 to 100.  The first step of the program has been completed, continue to select the second step.  Continue to step 3.  I entered 15, it does not exist. So it has been inserted into the 3 position of the sorted data.  If I entered 18, has been found, at the 3 position in Sorted Data.  After the program function is executed, enter 4 to exit. |
| 1. ****Summary****   1.Pay attention to the impact of the buffer on the input, you need to clear it with getchar().  2. Modular design is very important in programming. |
| **7.Reference**  **I have no reference.** |
| **Score：**  **Teacher（signature）：**  **Date:** |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Assessment Index | | level | | | | |
| A | B | C | D | E |
| Program Correctness | |  |  |  |  |  |
| Program Style& Interface | |  |  |  |  |  |
| Report Standardization | |  |  |  |  |  |
| Grade |  | | | | | |