Министерство образования и науки Российской Федерации

Севастопольский государственный университет

Кафедра ИС

Отчет

По дисциплине: “Операционные системы”

Лабораторная работа №3

“Информационные структуры ОС. Таблица дескрипторов файлов”

Вариант 3

Выполнил:

ст.гр. ИС/б-22

Долженко И.А.

Проверила:

Шалимова Е.М.

Севастополь

2019

1 ЦЕЛЬ РАБОТЫ

Изучение информационных структур ОС, получение навыков обработки массивов данных и отладки программ циклической структуры.

2 ЗАДАНИЕ

Разработать функцию сортировки таблицы дескрипторов файлов по заданному ключу. Таблицу дескрипторов представить массивом структур. Дескриптор должен содержать имя файла (не более 8 символов), тип файла (не более 3 символов), дату создания (в формате чч.мм.гг), количество обращений (целое число), размер файла (целое число), время последней модификации(в формате час.мин). В главной программе предусмотреть ввод и вывод исходных данных и результатов, а также обращение к функции сортировки, глобальные переменные не использовать.

Тип ключа: дата создания

Метод сортировки: сортировка обменом

3 ТЕКСТ ПРОГРАММЫ

#include <string.h>

#include <iostream>

#include <iomanip>

using namespace std;

struct file\_mas

{

char name[9], type[4];

int data\_year, data\_mon, data\_day;

int count\_obr;

int size;

int time\_hour, time\_min;

} file;

void vvod(struct file\_mas \*file, int N);

void vivod(struct file\_mas \*file, int N);

void sort(struct file\_mas \*file, int N);

int main()

{

setlocale(0,"RUS");

char c; int N;

cout<<"Введите размер массива структуры: "; cin>>N;

file\_mas \*file = new file\_mas[N];

vvod(file, N);

vivod(file, N);

sort(file, N);

vivod(file, N);

}

void vvod (struct file\_mas \*file, int N)

{

cout << "\n----------ВВОД ДАННЫХ----------" << endl;

for (int i = 0; i < N; i++)

{

cout << "Название файла: "; cin >> gets(file[i].name);

cout << "Тип файла: "; cin >> gets(file[i].type);

cout << "Ввод даты создания." << endl;

cout << " День: "; cin >> file[i].data\_day;

cout << " Месяц: "; cin >> file[i].data\_mon;

cout << " Год: "; cin >> file[i].data\_year;

cout << "Кол-во обращений: "; cin >> file[i].count\_obr;

cout << "Размер файла: "; cin >> file[i].size;

cout << "Время последней модификации." << endl;

cout << " Часы: "; cin >> file[i].time\_hour;

cout << " Минуты: "; cin >> file[i].time\_min;

cout << "--------------------------------" << endl;

}

}

void vivod (struct file\_mas \*file, int N)

{

cout << "\n|------------------------ВЫВОД ДАННЫХ------------------------|" << endl;

cout << "|------------------------------------------------------------------------|" << endl;

cout <<"| Название | Тип | Дата | Обращения | Размер | Время модификации |"<<endl;

cout << "|------------------------------------------------------------------------|" << endl;

for (int i = 0; i < N; i++)

{

cout<< setw(10) << file[i].name

<< setw(6) << file[i].type

<< setw(6) << file[i].data\_day << "." << file[i].data\_mon << "."

<< file[i].data\_year

<< setw(13) << file[i].count\_obr

<< setw(9) << file[i].size

<< setw(17) << file[i].time\_hour << ":" << file[i].time\_min << endl;

cout << "-------------------------------------------------------------------------" << endl;

}

}

void sort (struct file\_mas \*file, int N)

{

for (int i = 0; i< N-1; i++)

for (int j = 0; j<N-i-1; j++)

if (file[j].data\_year > file[j+1].data\_year ||

file[j].data\_year == file[j+1].data\_year &&

file[j].data\_mon > file[j+1].data\_mon ||

file[j].data\_year == file[j+1].data\_year &&

file[j].data\_mon == file[j+1].data\_mon &&

file[j].data\_day > file[j+1].data\_day)

{

char s1[9];

strcpy(s1, file[j].name);

strcpy(file[j].name, file[j+1].name);

strcpy(file[j+1].name, s1);

char s2[4];

strcpy(s2, file[j].type);

strcpy(file[j].type, file[j+1].type);

strcpy(file[j+1].type, s2);

int temp1= file[j].data\_year;

file[j].data\_year= file[j+1].data\_year;

file[j+1].data\_year = temp1;

int temp2= file[j].data\_mon;

file[j].data\_mon= file[j+1].data\_mon;

file[j+1].data\_mon = temp2;

int temp3= file[j].data\_day;

file[j].data\_day= file[j+1].data\_day;

file[j+1].data\_day = temp3;

int temp4= file[j].count\_obr;

file[j].count\_obr= file[j+1].count\_obr;

file[j+1].count\_obr = temp4;

int temp5= file[j].size;

file[j].size= file[j+1].size;

file[j+1].size = temp5;

int temp6= file[j].time\_hour;

file[j].time\_hour= file[j+1].time\_hour;

file[j+1].time\_hour = temp6;

int temp7= file[j].time\_min;

file[j].time\_min= file[j+1].time\_min;

file[j+1].time\_min = temp7;

}

}

4 ТЕСТИРОВАНИЕ

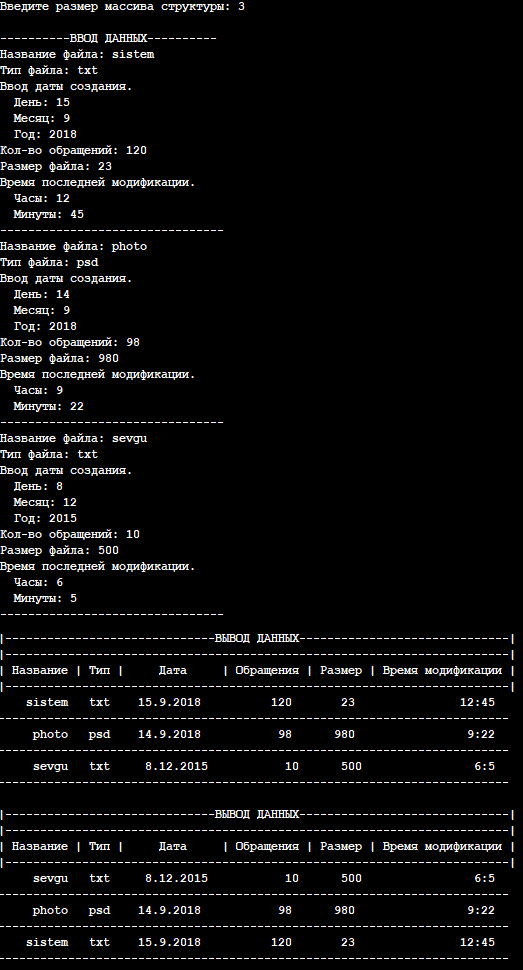


Рисунок 1 – Тест №1

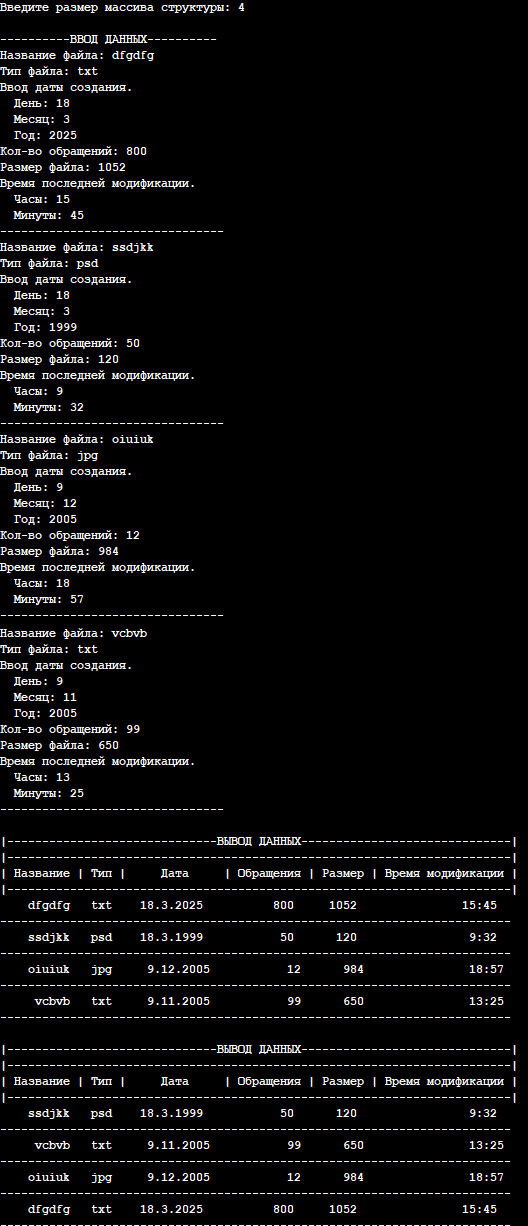


Рисунок 2 – Тест №2

ВЫВОД

В ходе выполнения лабораторной работы была разработана функция сортировки таблицы дескрипторов файлов по заданному ключу. Таблица дескрипторов представлена массивом структур. В главной программе предусмотрен ввод и вывод исходных данных и результатов, а также обращение к функции сортировки.