Министерство науки и высшего образования Российской Федерации

Пензенский государственный университет

Кафедра «Вычислительная техника»

**ОТЧЁТ**

По лабораторной работе №10

По курсу программирование

На тему «Файлы»

Выполнили

Студенты группы 24ВВВ3:

Давыдов А.Р.

Буров А.О.

Алмакаев В.В.

Приняли:

К.т.н. доцент Юрова О.В.

К.т.н. доцент Федюнин Р.Н.

Пенза 2024

**Название**

Файлы

**Цель работы**

Изучение способов описания файлов и основных принципов работы с файлами в различных режимах: создания, модификации, добавления и чтения.

**Лабораторное задание**

1. Лабораторная работа № 10 выполняется на основе работы № 8.

2. В задание №6 лабораторной работы добавить работу с файлами:

а) в функции ввода исходные данные записывать в файл;

б) в функциях обработки и вывода данные читать из файла, результат сохранять в файл;

в) для второго задания лабораторной работы №6 данные дописывать в файл с исходными данными для первого задания;

г) результаты выполнения и первого, и второго задания тоже должны быть сохранены в одном файле.

3. Выполнить программу на компьютере и оценить правильность ее работы.

4. Вывести на печать содержимое файла исходных данных и результатов работы программы.

**Вариант 4**

а). Вычислить среднее арифметическое массива М(10).

б). Из двух массивов A(5) и B(6) сформировать третий массив C(11) и отсортировать его по возрастанию.

**Описание метода решения задачи**

Программа предназначена для ввода трех массивов целых чисел, вычисления среднего арифметического элементов одного из массивов и объединения двух массивов в один, который затем сортируется и выводится на экран.

**Листинг**

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdio.h>

#include <stdlib.h>

#include <locale.h>

#include <time.h>

#define SIZE\_M 10

#define SIZE\_A 5

#define SIZE\_B 6

#define SIZE\_C (SIZE\_A + SIZE\_B)

void inputArray(int\* arr, int size, const char\* filename) {

FILE\* file;

fopen\_s(&file, filename, "a"); if (!file) {

perror("Unable to open file");

exit(EXIT\_FAILURE);

}

for (int i = 0; i < size; i++) {

int value = rand() % 100;

fprintf(file, "%d ", value);

}

fprintf(file, "\n");

fclose(file);

}

float calculateAverage(const char\* filename, int\* arr, int size) {

FILE\* file;

fopen\_s(&file, filename, "r");

if (!file) {

perror("Unable to open file");

exit(EXIT\_FAILURE);

}

float sum = 0;

for (int i = 0; i < size; i++) {

fscanf\_s(file, "%d", &arr[i]);

sum += arr[i];

}

fclose(file);

return sum / size;

}

void combineArrays(const char\* filename, int\* A, int\* B, int\* C) {

FILE\* file;

fopen\_s(&file, filename, "r");

if (!file) {

perror("Unable to open file");

exit(EXIT\_FAILURE);

}

for (int i = 0; i < SIZE\_A; i++) {

fscanf\_s(file, "%d", &A[i]);

C[i] = A[i];

}

for (int i = 0; i < SIZE\_B; i++) {

fscanf\_s(file, "%d", &B[i]);

C[SIZE\_A + i] = B[i];

}

fclose(file);

}

void bubbleSort(const char\* filename, int\* arr, int n) {

FILE\* file;

fopen\_s(&file, filename, "r");

if (!file) {

perror("Unable to open file");

exit(EXIT\_FAILURE);

}

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

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

if (arr[j] > arr[j + 1]) {

int temp = arr[j];

arr[j] = arr[j + 1];

arr[j + 1] = temp;

}

}

}

fclose(file);

}

void writeResultsToFile(const char\* filename, float average, int\* C) {

FILE\* file = fopen(filename, "a");

if (file != NULL) {

fprintf(file, "Среднее арифметическое массива M: %.2f\n", average);

fprintf(file, "Отсортированный массив C:\n");

for (int i = 0; i < SIZE\_C; i++) {

fprintf(file, "%d ", C[i]);

}

fprintf(file, "\n");

fclose(file);

}

else {

printf("Ошибка открытия файла для записи.\n");

}

}

int main() {

setlocale(LC\_ALL, "");

srand(time(NULL));

int M[SIZE\_M], A[SIZE\_A], B[SIZE\_B], C[SIZE\_C];

float average = 0.0;

char fileName1[100], fileName2[100], fileName3[100];

int currentIndex = 0;

printf("Введите имя файла для данных массива M: ");

scanf("%s", fileName1);

printf("Введите имя файла для данных массивов A, B: ");

scanf("%s", fileName2);

printf("Введите имя файла для вывода: ");

scanf("%s", fileName3);

FILE\* file1;

fopen\_s(&file1, fileName1, "w");

fclose(file1);

FILE\* file2;

fopen\_s(&file2, fileName2, "w");

fclose(file2);

FILE\* file3;

fopen\_s(&file3, fileName3, "w");

fclose(file3);

inputArray(M, SIZE\_M, fileName1);

average = calculateAverage(fileName1, M, SIZE\_M);

inputArray(A, SIZE\_A, fileName2);

inputArray(B, SIZE\_B, fileName2);

combineArrays(fileName2, A, B, C);

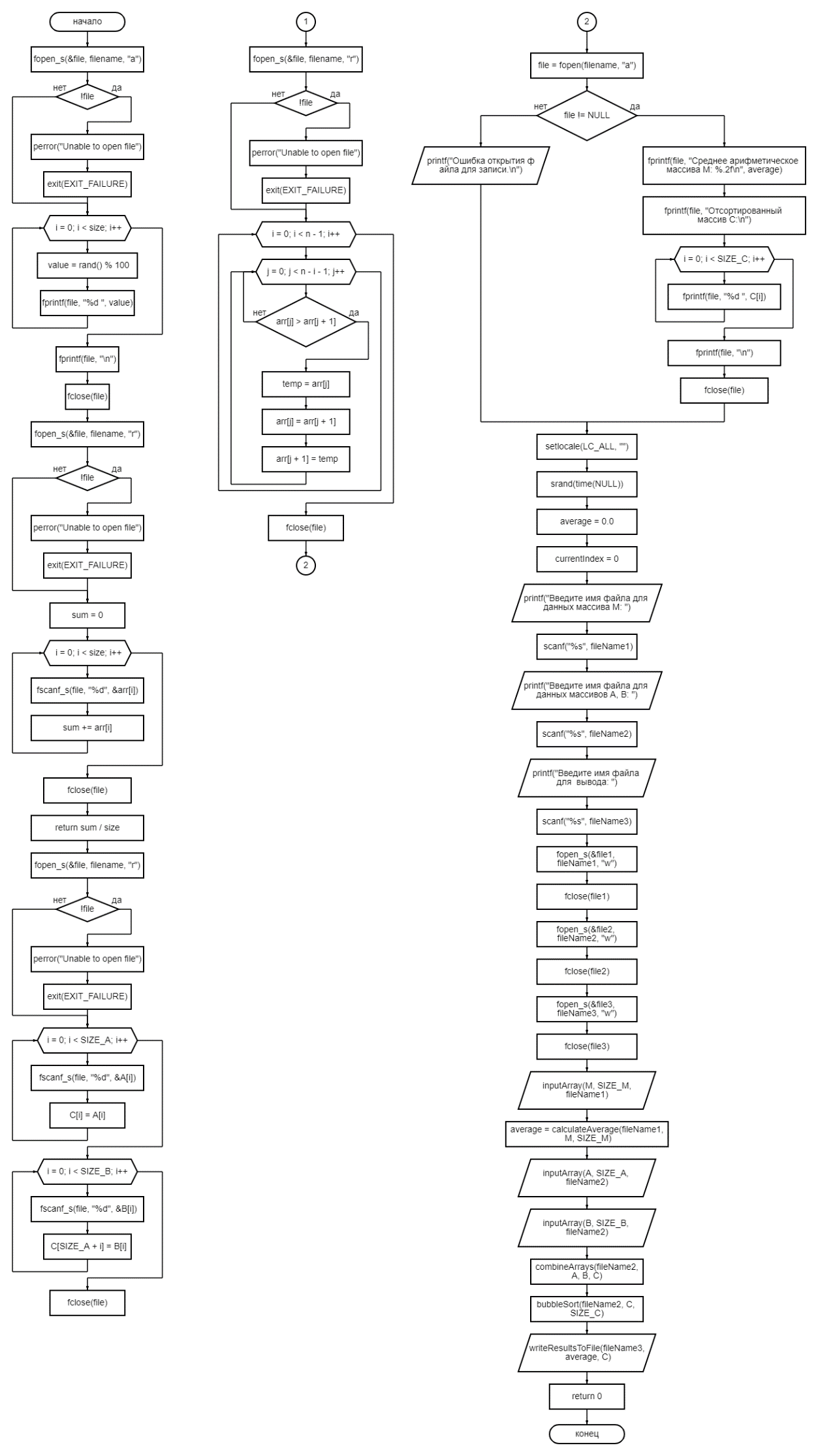
bubbleSort(fileName2, C, SIZE\_C);

writeResultsToFile(fileName3, average, C);

return 0;

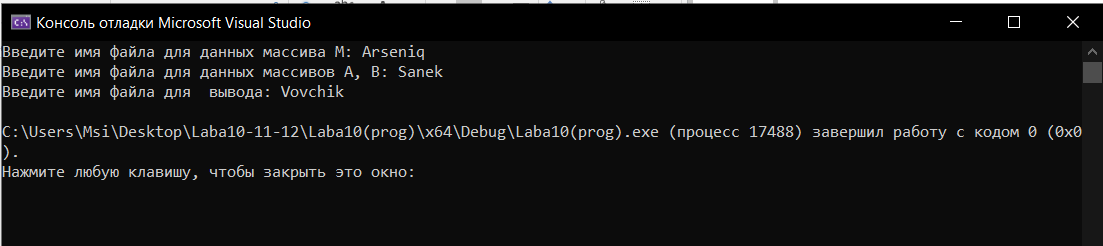
}

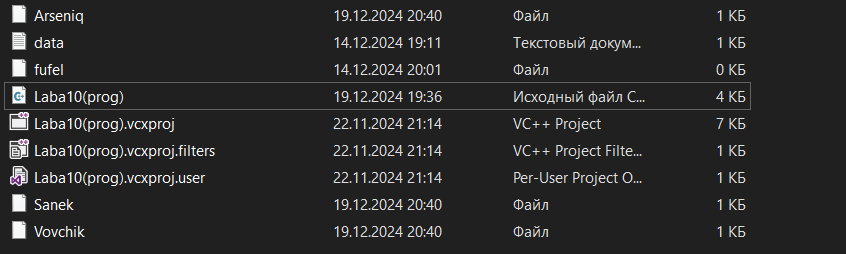
**Схема работы программы**



**Результаты работы программы**

Результат работы программы показан на рисунке 2.

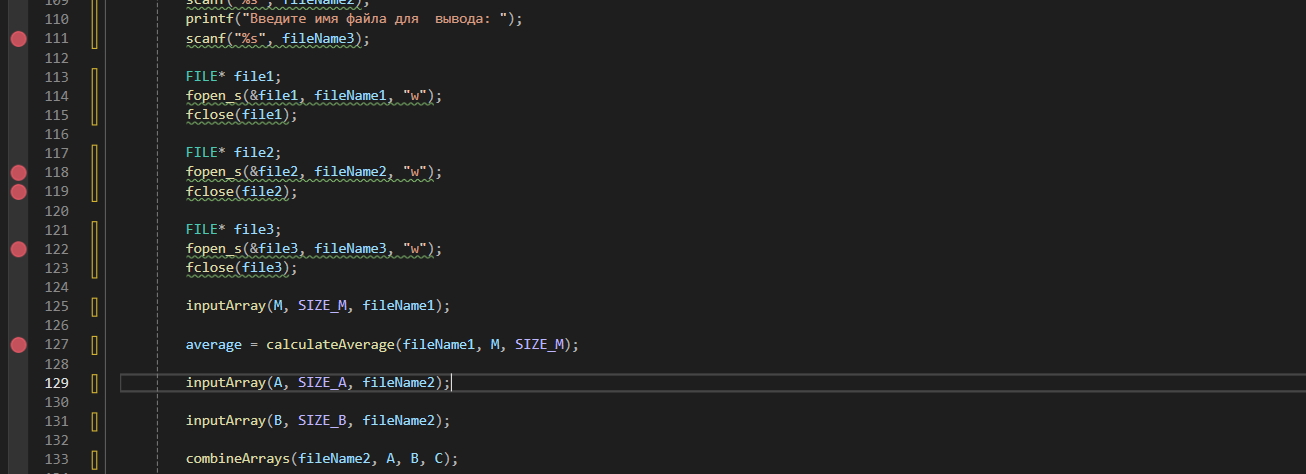




**Рисунок 2**

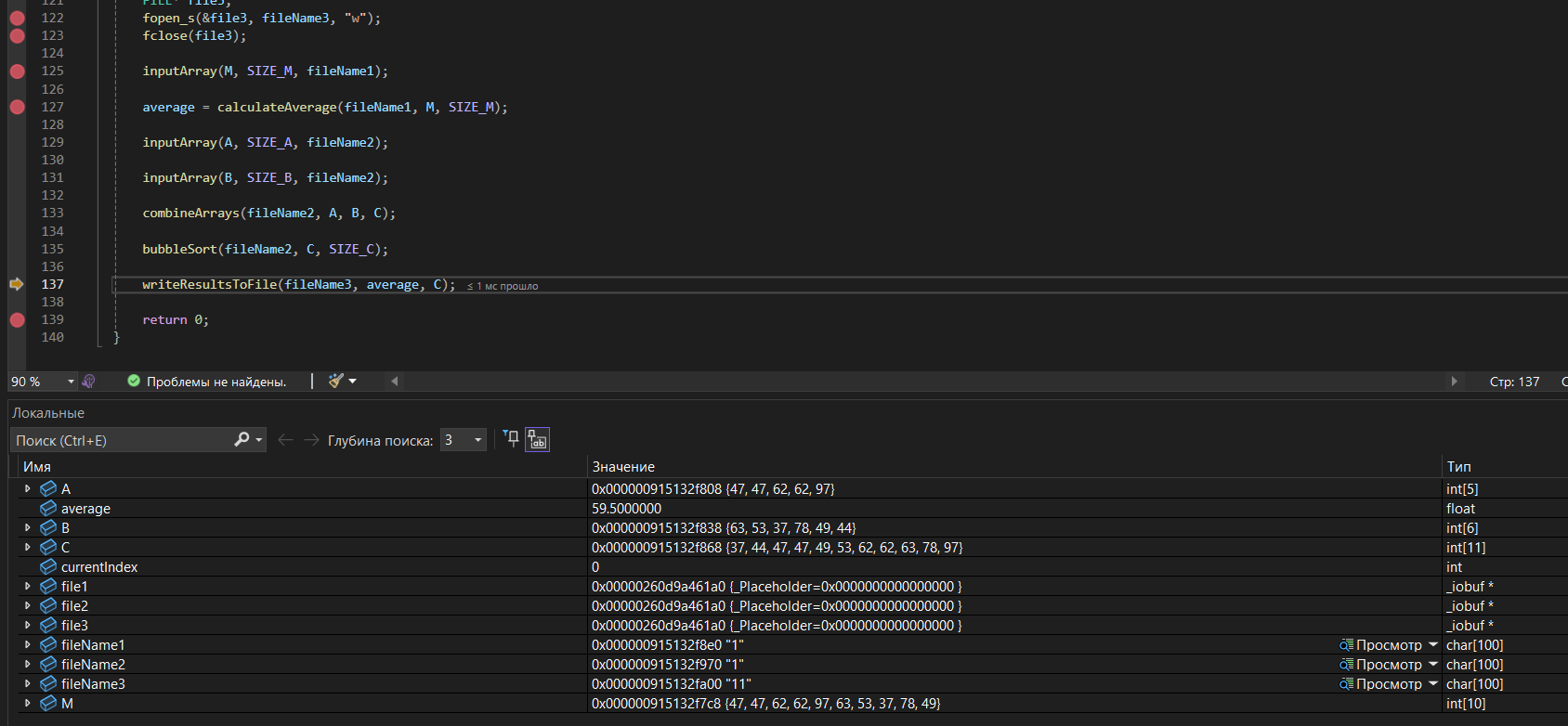
**Трассировка программы**

На рисунке 3 показан протокол трассировки



**Рисунок 3**

На рисунке 4 показан протокол трассировки после вывода двумерного массива.



**Рисунок 4**

**Вывод:**

В ходе выполнения лабораторной работы была разработана программа которая ищет элемент среди элементов массива, который оканчивается на 0.

Программа, которая ищет максимум и минимум среди сумм строк двумерного массива. Все обращения к массивам выполнены при помощи указателей. Результаты работы программы совпали с ожиданиями.

изучили способы описания файлов и основных принципов работы с файлами в различных режимах: создания, модификации, добавления и чтения.