Министерство образования и науки Российской Федерации

Пензенский государственный университет

Кафедра «Вычислительная техника»

**ОТЧЕТ**

по лабораторной работе №12

по курсу «Программирование»

на тему «Многомодульные программы»

Вариант №1

Выполнили:

студенты группы 22ВВ3

Кочетков А.М.

Кузьмин Д. В.

Матюшин К. М.

Принял:

Юрова О.В.  
Патунин Д.В.

Пенза 2022

### Название

Многомодульные программы.

### Цель работы

Изучение принципов написания многомодульных программ и приобретение навыков в написании таких программ и работе с ними.

### Лабораторное задание

Данная лабораторная работа выполняется на основе лабораторного задания к работе № 8 "Функции".

1. Выбрать вариант задания в соответствии с номером бригады.

2. Каждая функция 8-ой лабораторной работы должна быть помещена в отдельный файл. Далее в среде необходимо создать проект, в который включаются все файлы с функциями.

3. Выполнить программу на компьютере и оценить правильность ее работы на различных наборах данных.

### Описание метода решения задачи

Результаты Декларация функций и объявление глобальных переменных записывается в отдельном файле center.h. Каждая функция также записывается в отдельный .c файл.

### Листинг

#### Файл Lab12.c

#include "center.h"

#include "create\_array.c"

#include "display\_array.c"

#include "calculations.c"

#include "output.c"

#include <stdio.h>

void main() {

create\_array(11, &array1[0], 20, 1);

create\_array(11, (int \*)array2, 6, 8);

display\_array(&array1[0], 20, 1);

display\_array((int \*)array2, 6, 8);

calculations(&array1[0], (int \*)array2, &ans1, &ans2, k);

output(ans1, ans2);

}

#### Файл center.h

void create\_array(int seed, int \*pa, int w, int h);

void display\_array(int \*pa,int w, int h);

void calculations(int \*pa, int \*pb, int \*ans1, int \*ans2, int k);

void output(int ans1, int ans2);

int array1[20], array2[6][8], ans1, ans2, k;

#### Файл create\_array.c

#include "center.h"

#include <stdlib.h>

void create\_array(int seed, int \*pa, int w, int h) {

srand(seed);

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

for (int j=0; j<w; j++) {

\*pa = rand()%(200)-100;

pa++;

}

}

}

#### Файл display\_array.c

#include <stdio.h>

void display\_array(int \*pa,int w, int h) {

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

printf("\nrow %d:", i+1);

for (int j=0; j<w; j++) {

printf("%d, ", \*pa);

pa++;

}

}

printf("\n");

}

#### Файл calculations.c

void calculations(int \*pa, int \*pb, int \*ans1, int \*ans2, int k) {

for (int i=0; i<20; i++) {

if (i%2 == 0) {

\*ans1 += \*pa;

}

else {

\*ans1 -= \*pa;

}

pa++;

}

for (int i=0; i<6; i++) {

for (int j=0; j<8; j++) {

if ((i+j) == k) {

\*ans2 += \*pb;

}

pb++;

}

}

}

#### Файл output.c

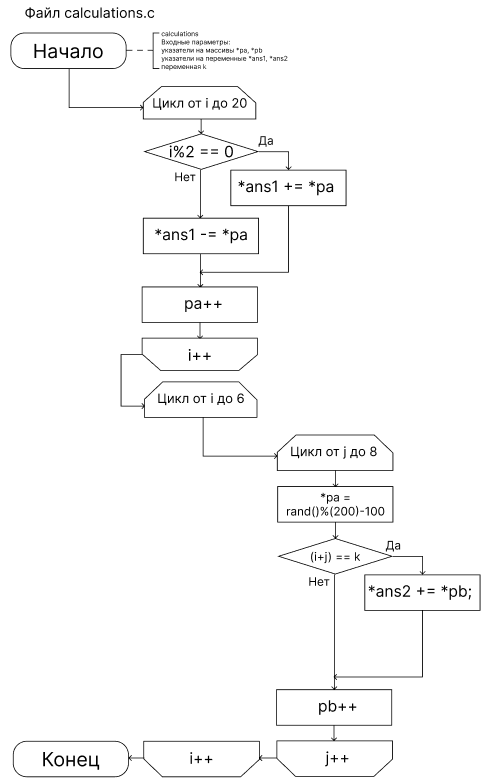
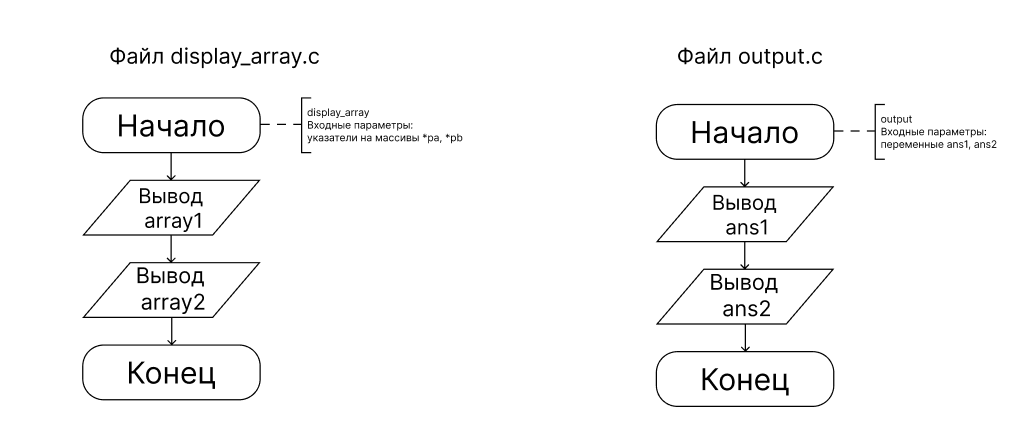
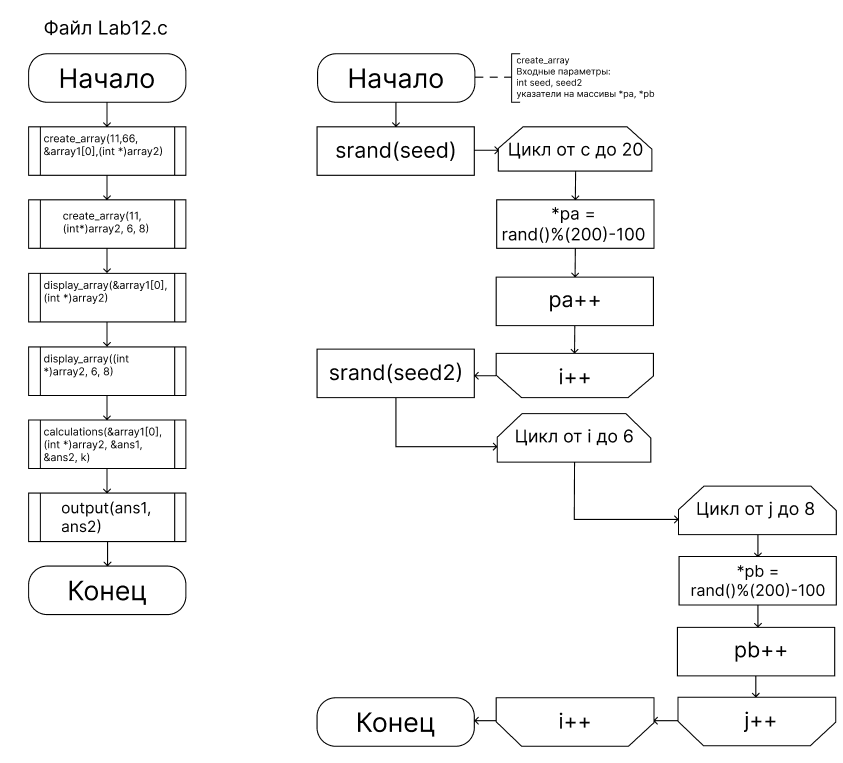
void output(int ans1, int ans2) {

printf("Specified sum of the elements of the array is: %d\nSum of matrix elements whose sum of indexes is equal to a constant: %d", ans1, ans2);

}

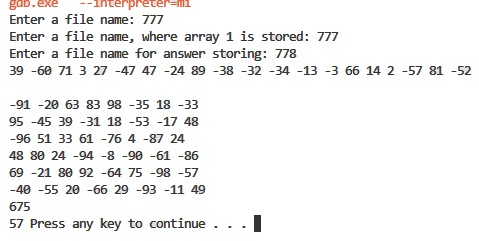
Пояснительный текст к программе

Переменные k, a, b, pa, pb взяты из прошлой работы. Sum и ans – переменный для ответов.



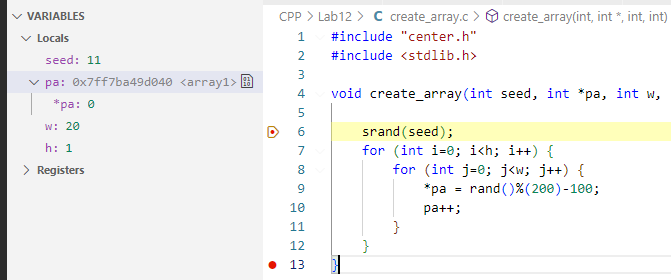
### Результаты работы программы

Результаты работы программы показан на рисунке 1.

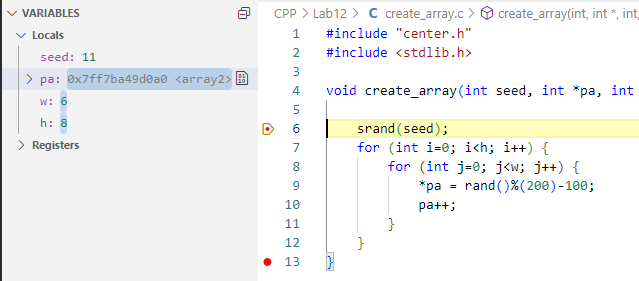
  
**Рисунок 1 — Результаты работы программы**

### Протокол трассировки программы

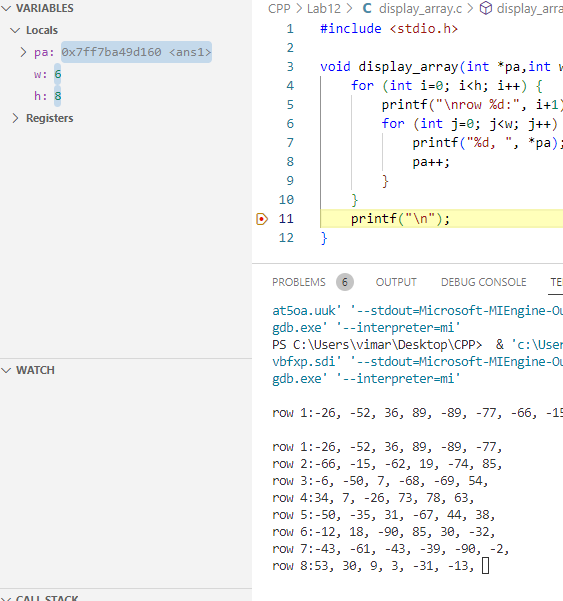
На рисунке 3 показан протокол трассировки программы



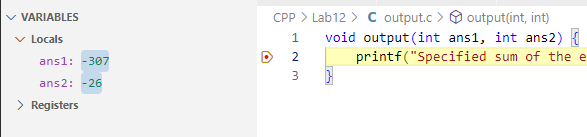
**Рисунок 2.1**



**Рисунок 2.2**



**Рисунок 2.3**



**Рисунок 2.4**

Алгоритмы выполнили вывод данных верно.

### Вывод

В ходе выполнения лабораторной работы была написана программа, осуществляющая создание массивов, проводящая заданные вычисления, выводящая результаты на экран.

Результат работы программы совпал с результатом трассировки и ожидаемыми данными, следовательно, программа работает без ошибок.

Получен опыт в создании проектов в среде MicrosoftVisualStudio, работе с многомодульными программами, в их написании.