Пензенский государственный университет

Кафедра "Вычислительная техника"

**ОТЧЕТ**

по лабораторной работе №8

по курсу «Программирование»

на тему "Функции"

Выполнили:

Студенты группы 22ВВ2

Ипполитов И. Д.

Горбатов К. В.

Приняли:

Юрова О. В.

Патунин Д. В.

Пенза 2022

**Название**Функции

**Цель работы**Изучение правил составления и написания функций и интерфейса функционального модуля.

**Лабораторное задание**

1. a). Задан массив А(20). Определить знакопеременную сумму а1– а2 + а3 – а4 + ….

б). Просуммировать элементы матрицы Х(6,8), сумма индексов которых равна заданной константе К.

2. Составить программу, которая выполняет следующие действия:

а) ввод исходных массивов;

б) вывод исходных массивов;

в) обработку массивов в соответствии с заданием;

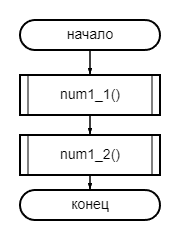
г) вывод результатов с соответствующими комментариями.

Пункты а), б), в), г) оформить в виде функций. Глобальные данные в программе не использовать.

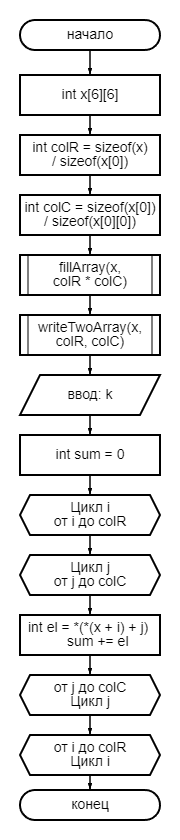
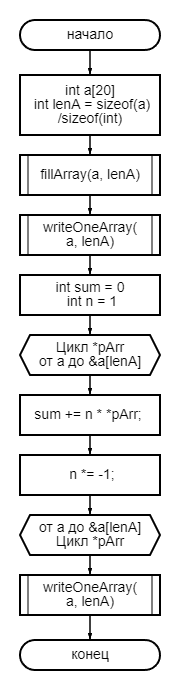
**Ход работы:**

1. **Схема программы**

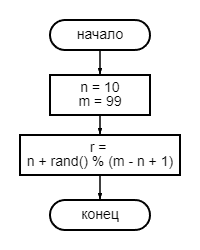
* **main.c**

****

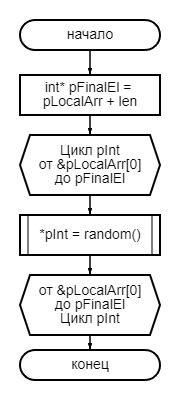
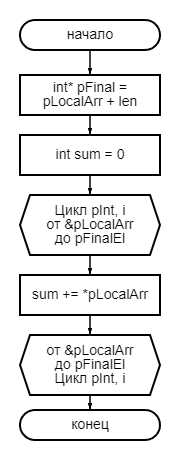
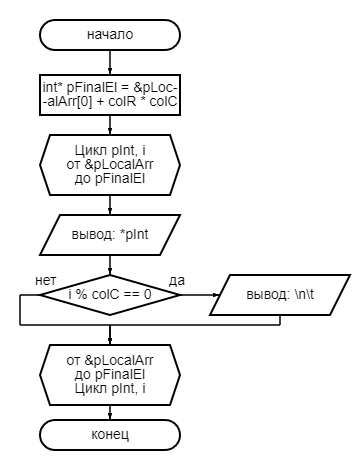
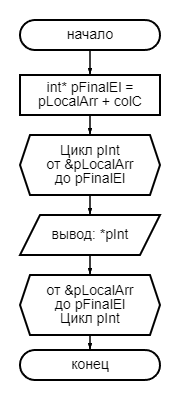
* **var1.c**

****

* **func.h**



* **arrayWork.h**



1. **Код программы**

* **main.c**

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdio.h>

#include "arrayWork.h"

int main()

{

srand(time(NULL));

printf("==Var1==\n\n");

num1\_1();

printf("\n-------------\n");

num1\_2();

return 0;

}

* **var1.c**

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdio.h>

int num1\_1()

{

int a[20];

int lenA = sizeof(a) / sizeof(int);

fillArray(a, lenA);

writeOneArray(a, lenA);

int sum = 0;

int n = 1;

for (int\* pArr = a; pArr < &a[lenA]; pArr++)

{

sum += n \* \*pArr;

n \*= -1;

}

printf("\nSum: %d", sum);

return 0;

}

int num1\_2()

{

int x[6][6];

int colR = sizeof(x) / sizeof(x[0]);

int colC = sizeof(x[0]) / sizeof(x[0][0]);

//int\* finEl = \*x + colC \* colR - 1;

fillArray(x, colR \* colC);

writeTwoArray(x, colR, colC);

int k = 0;

printf("Insert dop num: ");

scanf("%d", &k);

int sum = 0;

printf("Elements which indexes sum equals '%d': ", k);

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

{

for (int j = 0; j < colC; j++)

{

if (i + j == k)

{

int el = \*(\*(x + i) + j);

sum += el;

printf("%d ", el);

}

}

}

printf("\nSum: %d", sum);

return 0;

}

* **func.h**

#pragma once

#include <stdlib.h>

#include <time.h>

// generate random bumber

int random()

{

// числа генерируются от 10 до 99

int n = 10;

int m = 99;

// генерируется рандомное число

int r = n + rand() % (m - n + 1);

return r;

}

* **arrayWork.h**

#pragma once

#include "func.h"

#include <stdlib.h>

#include <stdio.h>

int fillArray(int\* pLocalArr, int len)

{

int\* pFinalEl = pLocalArr + len;

for (int\* pInt = &pLocalArr[0]; pInt < pFinalEl; pInt++)

{

\*pInt = random();

}

return 0;

}

int writeOneArray(int\* pLocalArr, int colC)

{

int\* pFinalEl = pLocalArr + colC;

printf("Array:\n\t");

for (int\* pInt = pLocalArr; pInt < pFinalEl; pInt++)

{

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

}

printf("\n");

return 0;

}

int writeTwoArray(int\* pLocalArr, int colR, int colC)

{

int\* pFinalEl = &pLocalArr[0] + colR \* colC;

printf("Array:\n\t");

for (int\* pInt = &pLocalArr[0], i = 1; pInt < pFinalEl; pInt++, i++)

{

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

if (i % colC == 0)

printf("\n\t");

}

printf("\n");

return 0;

}

// find sum of array

int arraySum(int\* pLocalArr, int len)

{

int\* pFinal = pLocalArr + len;

int sum = 0;

for (pLocalArr; pLocalArr < pFinal; pLocalArr++)

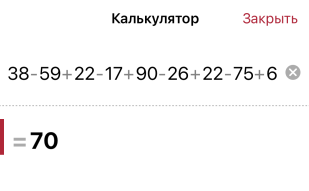
{

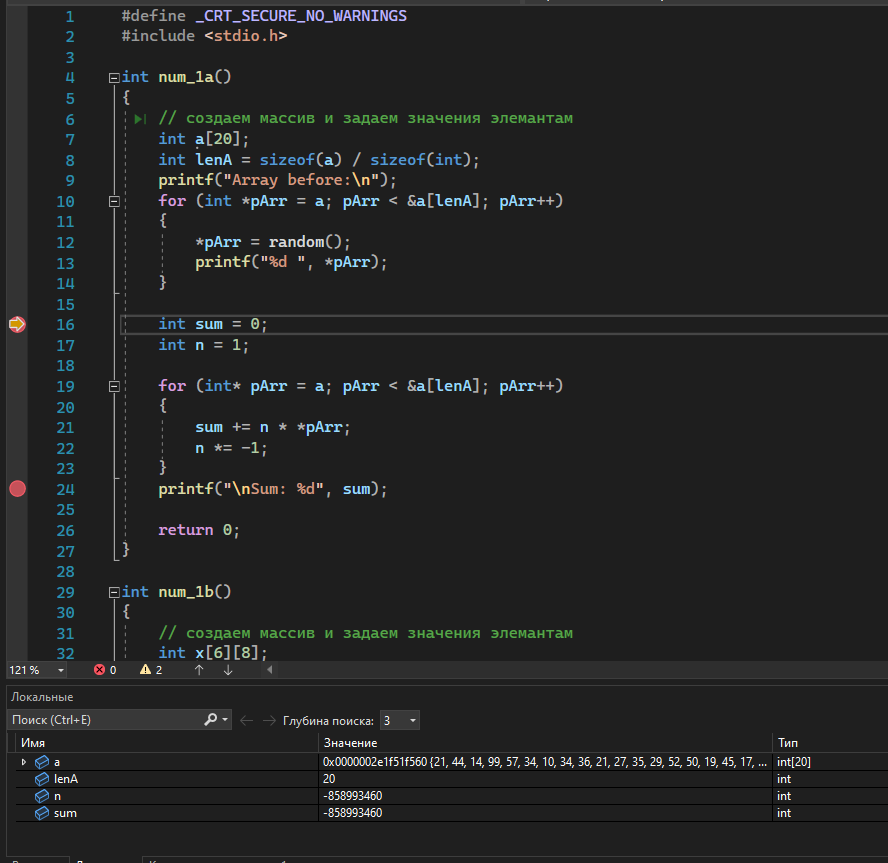
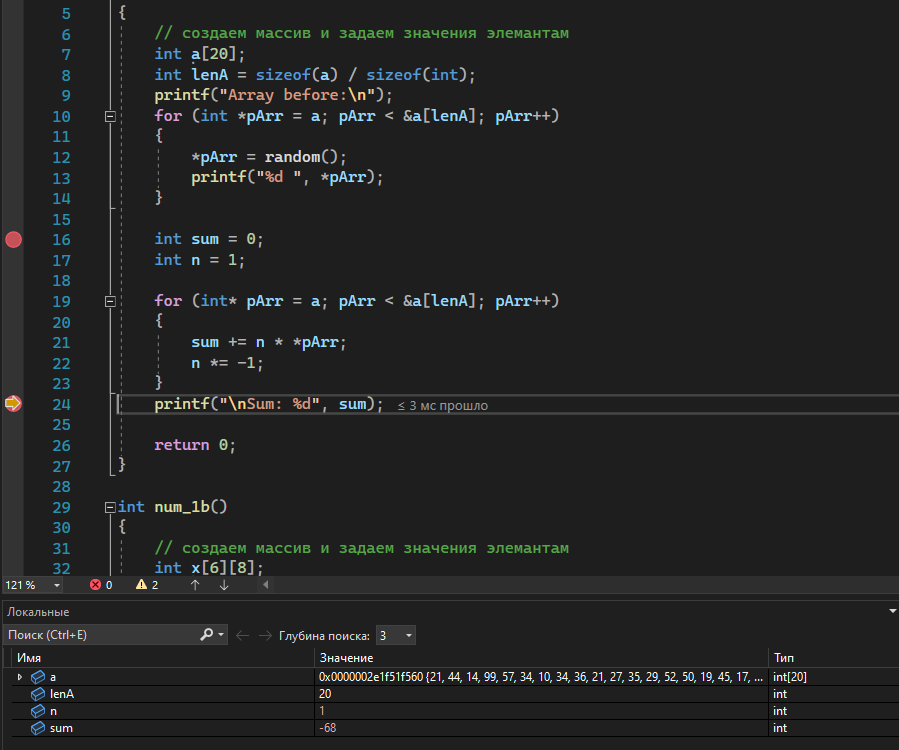
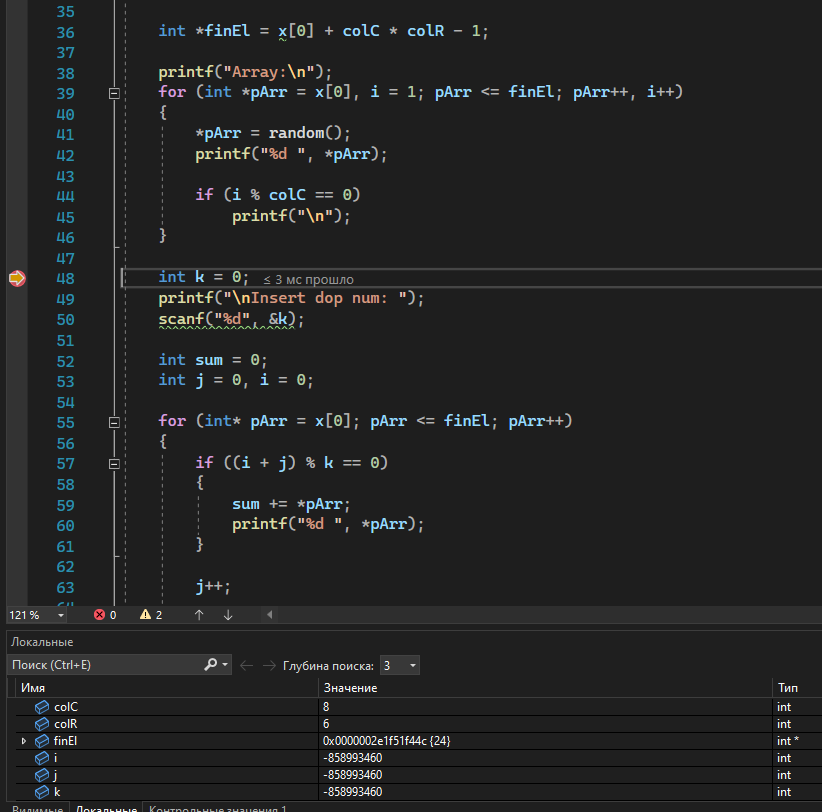
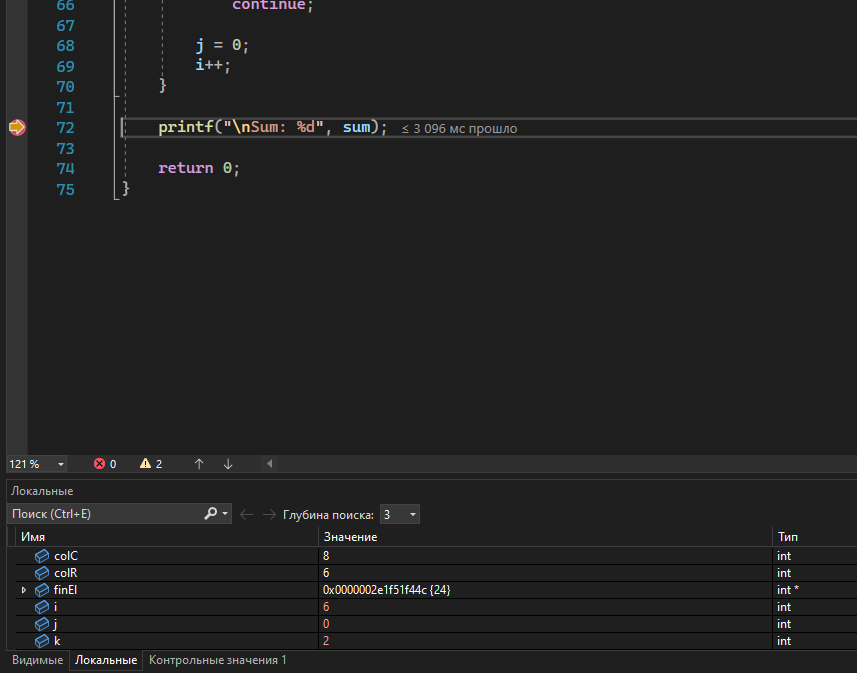
sum += \*pLocalArr;

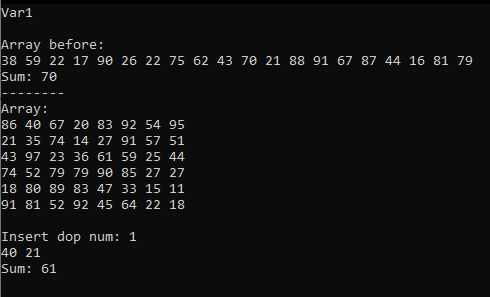
}

return sum;

}

1. **Ручной счет**
   1. 
   2. Дополнительное число k = 1. 1+0 = 0, 0+1 = 1  
      Подходят элементы, значение которых рано 40 и 21   
      Сумма равна 61
2. **Пояснительный текст к программе**
3. Находится длина массива A. Следом каждый элемент заполняется случайным числом. Потом в цикле элементы массива складываются друг с другом, изменяя знак каждую итерацию
4. Находим размерность двумерного массива X. Следом каждый элемент заполняется случайным числом. Потом запрашивается дополнительное число k и проверяется сумма индексов каждого элемента. Если сумма индексов равна числу k, то этот элемент добавляется в сумму.
5. **Трассировка**

1. **Результат работы программы**

**Вывод:**Мы изучили правила составления и написания функций и интерфейса функционального модуля. Благодаря этому мы сделали программу в который используются полученные нами навыки по созданию функций.