**ЛАБОРАТОРНАЯ РАБОТА №2**

**СОЗДАНИЕ МНОГОФАЙЛОВЫХ ПРОЕКТОВ**

**Цель**:

Целью данной лабораторной работы освоить на практике вызов функции с использованием указателей..

**Задания:**

1. Напишите программу, которая вызывает различные виды функции в зависимости от заданного условия. Можно использовать примеры выше.

2. Напишите программу с использованием некой универсальной функции, которая в качестве возвращаемого значения возвращает указатель на функцию. Эта функция в зависимости от исходного массива должна соответствующим образом преобразовать массив. Также в функции main() должен быть объявлен указатель. В теле программы указателю на функцию должно присваиваться значение. Исходные данные - сгенерировать целочисленный массив из случайных чисел. Запрограммировать следующие действия в виде функций: • если сумма элементов в массиве равна его первому элементу, то необходимо инвертировать массив

• если сумма элементов в массиве больше его первого элемента, то необходимо расположить его элементы в неубывающем порядке

• если сумма элементов массива меньше его первого элемента, то необходимо расположить его элементы в

невозрастающем порядке

**Задание №1**

#include "pch.h"

#include "windows.h"

#include <iostream>

#include <string>

using namespace std;

double sum(double a, double b)

{

return a + b;

}

double multiplication(double a, double b)

{

return a \* b;

}

double dif(double a, double b)

{

return a - b;

}

double division(double a, double b)

{

return a / b;

}

double(\*operation)(double a, double b);

int main()

{

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

system("color 0A");

double a, b;

cout << "Введите два числа" << endl;

cin >> a >> b;

int z;

cout << "1-разность 2-сумма 3-произвдение 4-деление" << endl;

cin >> z;

switch (z)

{

case 1:

operation = dif;

cout << operation(a, b) << endl;

break;

case 2:

operation = sum;

cout << operation(a, b) << endl;

break;

case 3:

operation = multiplication;

cout << operation(a, b) << endl;

break;

case 4:

if (b != 0)

{

operation = division;

cout << operation(a, b) << endl;

}

else

{

cout << "error" << endl;

}

default:

break;

}

system("pause");

return 0;

}

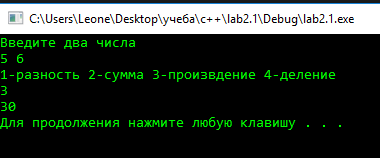


Рис.1

**Задание №2**

#include "pch.h"

#include "windows.h"

#include <iostream>

#include <string>

#include <ctime>

using namespace std;

void invert(int \*arr, int size); // Инвертирует массив

void sortMinToMax(int \*arr, int size); // Сортирует массив в порядке неубывания

void sortMaxToMin(int \*arr, int size); // Сортирует массив в порядке невозрастания

//Возвращает функцию, которую надо выполнить

void(\*f(int \*arr, int size))(int \*arr, int size);

int main()

{

srand(time(0));

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

system("color 0A");

int n = 10;

int \*arr = new int[n];

cout << "Изначальный массив: ";

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

{

arr[i] = rand() % 100;

cout << arr[i] << " ";

}

f(arr, n)(arr, n);

cout << endl << "Обработанный массив: ";

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

{

cout << arr[i] << " ";

}

cout << endl;

system("pause");

return 0;

}

void invert(int \*arr, int size)

{

for (int i = 0; i < size / 2; i++)

{

int c = arr[i];

arr[i] = arr[size - 1 - i];

arr[size - 1 - i] = c;

}

}

void sortMinToMax(int \*arr, int size)

{

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

{

for (int j = i + 1; j < size; j++)

{

if (arr[j] < arr[i])

{

int c = arr[i];

arr[i] = arr[j];

arr[j] = c;

}

}

}

}

void sortMaxToMin(int \*arr, int size)

{

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

{

for (int j = i + 1; j < size; j++)

{

if (arr[j] > arr[i])

{

int c = arr[i];

arr[i] = arr[j];

arr[j] = c;

}

}

}

}

void(\*f(int \*arr, int size))(int \*arr, int size)

{

int sum = 0;

for (int i = 1; i < size; i++)

{

sum += arr[i];

}

if (sum == arr[0])

{

return invert;

}

else {

if (sum < arr[0])

{

return sortMaxToMin;

}

else

{

return sortMinToMax;

}

}

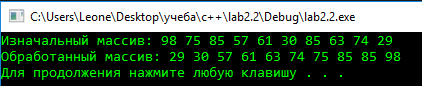
}****

Рис.2

**Вывод:** В ходе выполнения лабораторной работы №2 был реализован вызов функции с помощью указателей, а также показаны различные методы сортировки массива.