**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ**

**Федеральное государственное автономное образовательное учреждение высшего образования «Национальный исследовательский Нижегородский государственный университет им. Н.И. Лобачевского»**

**(ННГУ)**

**Институт информационных технологий, математики и механики**

Направление подготовки: «Фундаментальная информатика и информационные технологии»

Отчёт по лабораторной работе

**Сравнение элементов массивов**

Выполнил:

студент ИИТММ гр. 381906-2

Челебаев А.А.

Проверил:

ассистент каф. математического обеспечения и суперкомпьютерных технологий

Лебедев И.Г.

Нижний Новгород

2019 г.

Содержание

[Введение 3](#_Toc25003585)

[1. Постановка задачи 4](#_Toc25003586)

[2. Руководство пользователя 5](#_Toc25003587)

[3. Руководство программиста 6](#_Toc25003589)

[3.1. Описание структуры программы 6](#_Toc25003590)

[3.2. Описание структур данных 6](#_Toc25003591)

[3.3. Описание алгоритмов](#_Toc25003592) 6

[4. Эксперименты 7](#_Toc25003593)

[Заключение 9](#_Toc25003594)

[Литература 10](#_Toc25003595)

[Приложение 11](#_Toc25003596)

# Введение

Развитие современного общества предполагает широкое использование компьютерных и информационных технологий для работы с получаемой информацией.При решении задач, связанных с обработкой сразу множества значений данных одинакового типа, использование переменных с различными именами, не упорядоченных по адресам памяти, затрудняет программирование.В таких случаях крайне неудобно для хранения каждого значения заводить отдельную именованную переменную. Для такой ситуации в языке Си используют объекты, называемые массивами.

Массив – это набор однотипных объектов с общим именем; у каждого объекта есть свой уникальный номер, с помощью которого можно с ним работать, изменять, запрашивать значение.Каждый такой объект называется элементом массива. Номер элемента массива – это его индекс.Другими словами можно сказать, что массив представляет собой фиксированное количество упорядоченных однотипных компонент, снабженных индексами, т.е. является совокупностью конечного числа данных одного типа. В качестве элементов массива можно использовать любой тип данных, поэтому вполне правомерно существование массивов записей, массивов указателей, массивов строк, и т.д.

Массивы являются распространенным и полезным способом сохранения многих различных частей связанных данных. Массивы полезны при создании отсортированных и неотсортированных списков данных, при сохранении таблиц данных и для выполнения многих других задач. С понятием «массив» приходится работать и при решении научно-технических и экономических задач, связанных с обработкой совокупностей большого количества значений.

Массив является удобным способом хранения нескольких связанных элементов данных в едином контейнере для большего удобства и эффективности программирования. Массив позволяет сохранять и манипулировать многими элементами данных посредством единственной переменной. Кроме уменьшения общего числа различных имен переменных, которые необходимо отслеживать, другим основным преимуществом использования массивов является то, что можно использовать циклы для легкой обработки различных элементов массивов. Объединяя массивы и циклы, можно написать небольшое число операторов, которые обрабатывают большой объем данных. Выполнение тех же задач с использованием отдельных переменных может потребовать написания сотен операторов.

Данная работа демонстрирует принцип использования массива для выполнения конкретной задачи.

# Постановка задачи

Вывести на экран n случайных чисел в диапазоне от min до max, а также подсчитать сумму, которая получается следующим образом: все числа, номера которых совпадают с дробной частью какого-либо из исходных чисел, вычитаются, все остальные прибавляются. При этом n, min и max задаются пользователем непосредственно в процессе выполнения программы.

# Руководство пользователя

Использование программы пользователем:

Для того, чтобы запустить программу, перейдите по следующему пути:

**C:\Users\<Имя\_пользователя>\source\repos\<Название\_проекта>\Debug**,

Затем в строке состояния сотрите путь к каталогу и вместо него введите “**cmd**”. Откроется системная консоль. В командной строке введите название проекта и через пробел необходимое количество чисел, которое хотите увидеть, min и maxсоответственно, где minи max–диапазон значений элементов массива. После этого на экране будут напечатаны nэлементов массива и необходимая по условию задачи сумма (Рис.1).

# 

# Рисунок 1. Консольный интерфейс программы3. Руководство программиста

## 3.1. Описание структуры программы

1) Ввод данных;

2) Формирование массива на основе введённых данных;

3) Формирование массива дробных частей на основе массива чисел (Здесь также происходит формирование массива «флажков», нужного в дальнейшем для подсчёта суммы);

4) Подсчёт конечной суммы.

## 3.2. Описание структур данных

В работе используются переменные типа int(целочисленный)и double(тип вещественного числа с плавающей запятой).

Программа содержит главную функцию intmain(), являющуюся начальной точкой программы, а также функцию setlocale(), которая позволяет программе выводить надписи на русском языке.

К программе подключены следующие библиотеки:

stdio.h–нужна для ввода/вывода

stdlib.h – нужна для использования функции rand()

locale.h – позволяет сменить язык надписей

Помимо вышеперечисленных библиотек, программа содержит два цикла for,которые отвечают за формирование массивов чисел и их дробных частей и подсчёт конечной суммы соответственно, а также ветвление if, запускающее выполнение программы в случае, если в командную строку введено четыре аргумента.

Также в данной работе используется динамический массив. Динамическим называется массив, размер которого задаётся во время исполнения программы.

## 3.3. Описание алгоритмов

В начале программа считывает количество введённых аргументов в командной строке; в том случае, если их меньше четырёх, программа уведомляет пользователя об этом и просит его ввести корректное число аргументов. При этом пользователь должен ввести желаемое количество чисел, а также диапазон их значений от minдо max.

После этого происходит выделение памяти под динамические массивы massiv, drobи flag.

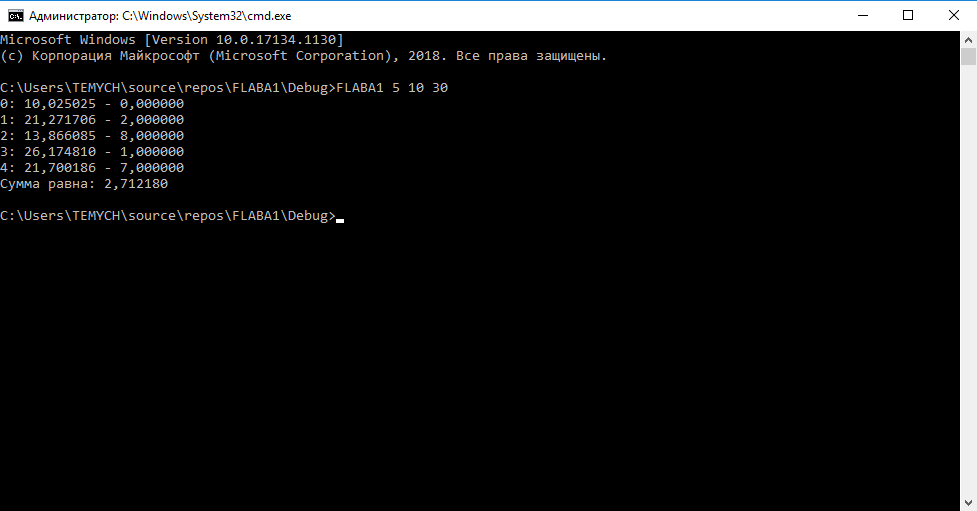
Затем в цикле forформируются массивы massivи flag, где в соответствие кадому элементу первого массива ставится «флажок» из второго. Помимо этого, здесь также формируется массив drob, содержащий дробные части чисел массива massiv. В этом же цикле задаётся система ветвлений if, позволяющая программе обрабатывать от 1 до 1000000 чисел.

Следом за этим циклом идёт другой цикл for, в котором происходит подсчёт конечной суммы по следующему принципу:

1. В начале цикла складываются все числа массива massiv;
2. Затем, если "флажок" числа ещё не использован (т.е. номер числа ещё не совпал с дробной частью какого-либо числа), вычитаются дважды (поскольку они уже прибавлены) те числа, у которых индекс совпадает с дробной частью какого-нибудь из чисел;
3. В конце программы выводится конечная сумма.

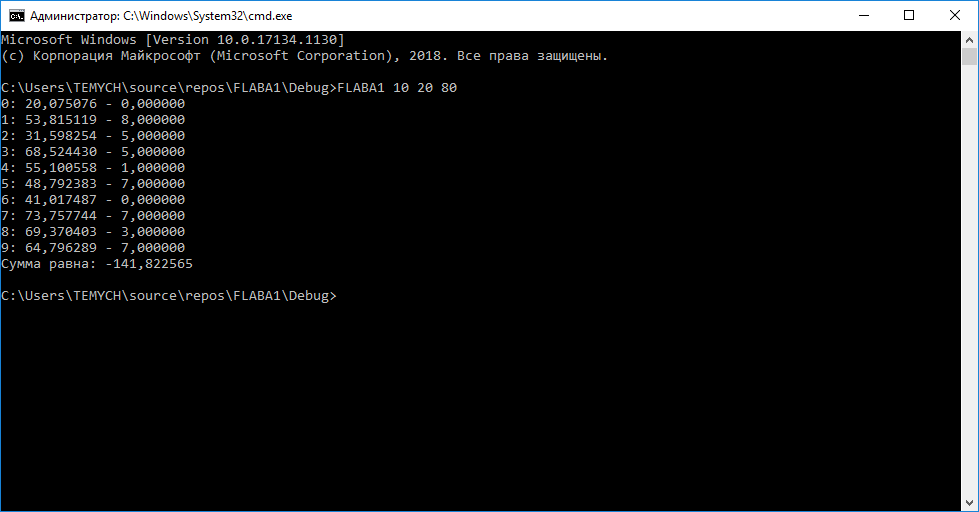
# 4. Эксперименты

Эксперимент №1 (Рис. 2)



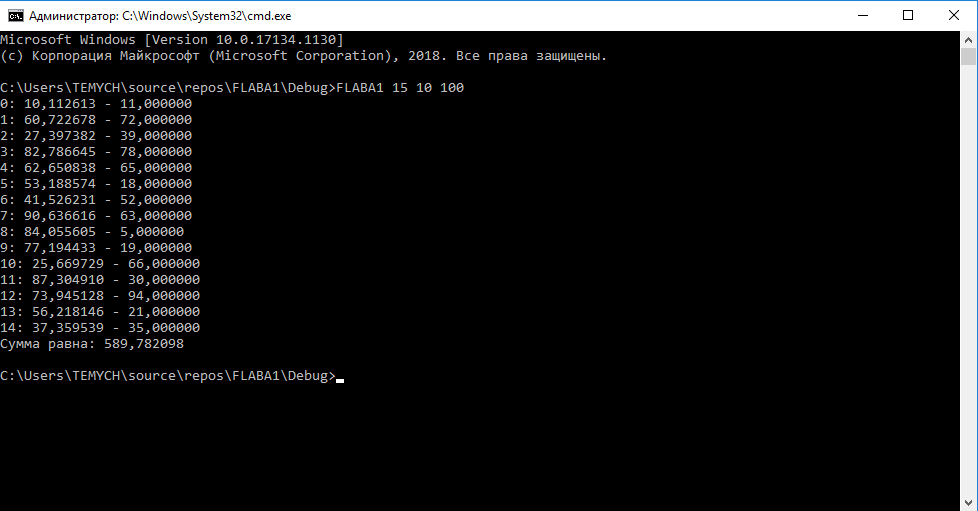
**Рисунок 2. Выполнение и результат эксперимента 1**

Эксперимент №2 (Рис. 3)



**Рисунок 3. Выполнение и результат эксперимента 2**

Эксперимент №3 (Рис.4)



**Рисунок 4. Выполнение и результат эксперимента 3**

# Заключение

Таким образом, в рамках данной лабораторной работы быласоздана программа, которая генерирует множество случайных чисел размера n в диапазоне (min, max) и после чего подсчитывает и выводит сумму, которая получается образом, описанным в разделе постановки задачи, а именно: все числа, номера которых совпадают с дробной частью одного из исходных чисел - вычитаются, все остальные прибавляются. В основе программы лежит динамический массив.

# Литература

1. Павловская Т.А. C/C++, Программирование на языке высокого уровня, 2003.
2. Официальный сайт Habr. – <https://habr.com/ru/post/339656/>
3. Официальный сайт Microsoft. – <https://docs.microsoft.com/ru-ru/cpp/c-language/main-function-and-program-execution?view=vs-2019>
4. Брайан Керниган, ДеннисРитчи. Язык программирования С. – Москва: Вильямс, 2015. – 304 с.
5. Грег Перри, Дин Миллер. Программирование на С для начинающих. – Москва: Эксмо, 2015. – 368 с.: ил.

# Приложение

***Код программы***

#include<stdio.h>

#include<stdlib.h>

#include<locale.h>

intmain(intargc, char\* argv[])

{

setlocale(LC\_ALL, "Rus");

inti = 0; //Индексмассивов

int n = 0; //Кол-во чисел

int x = 0; //Нужна для работы с массивом, ей мы будем присваивать случайные числа

int\* flag;

double\* massiv; //Указатель на массив изнач. чисел

double\* drob; //Указатель на массив дробных частей

double SUM = 0.0; //Конечная сумма, кот. нужно будет подсчитать

doublemax = 0; //Максимальное и миимальное значения

doublemin = 0;

//В том случае, если мы ввели в cmd меньше четырёх аргументов

if (argc< 4)

{

printf("Недостаточное количество аргументов, пожалуйста, введите минимум три аргумента\n");

return 1;

}

n = atoi(argv[1]);

min = atof(argv[2]);

max = atof(argv[3]);

//Выделяем память под нужные нам массивы

massiv = (double\*)malloc(n \* sizeof(double)); //Массивосновныхчисел

drob = (double\*)malloc(n \* sizeof(double)); //Массивдробныхчастей

flag = (int\*)malloc(n \* sizeof(int)); //Массив "маркеров"("Флажков")

for (i = 0; i < n; i++) //В этом цикле для каждого индекса ставится в соответствие "флажок", нужный для подсчёта конечной суммы

{

flag[i] = 0;

x = rand();

massiv[i] = (((double)x) / RAND\_MAX) \* (max - min) + min; //Здесь формируется массив из случайных чисел в диапазоне от min до max

drob[i] = (massiv[i] - (int)massiv[i]); //Здесь вычисляется дробная часть каждого из полученных чисел

if ((n > 0) && (n <= 10))

drob[i] = (drob[i] \* 10);

if ((n > 10) && (n <= 100))

drob[i] = (drob[i] \* 100);

if ((n > 100) && (n <= 1000))

drob[i] = (drob[i] \* 1000);

if ((n > 1000) && (n <= 10000))

drob[i] = (drob[i] \* 10000);

if ((n > 10000) && (n <= 100000))

drob[i] = (drob[i] \* 100000);

if ((n > 100000) && (n <= 1000000))

drob[i] = (drob[i] \* 1000000);

drob[i] = (int)drob[i]; //...а затем приводится к целочисленному виду, дабы мы могли сравнивать её с номерами чисел

printf("%d: %lf - %lf\n", i, massiv[i], drob[i]); //Здесь на экран выводятся индекс числа, само число и его дробная часть

}

for (i = 0; i < n; i++) //В этом цикле наша программа будет считать искомую сумму

{

SUM += massiv[i]; //Сначала по умолчанию складываем все числа

if (((int)drob[i] < n) && (flag[(int)drob[i]] == 0)) //Затем, если все дробные части чисел меньше кол-ва этих чисел и "флажок" ещё не использован, вычитаем дважды те числа, у кот. индекс совп. с дроб. частью какого-нибудь из чисел

{

SUM -= 2 \* massiv[(int)drob[i]]; //Вычитаем их дважды, поскольку мы их сначала уже прибавили

flag[(int)drob[i]] = 1; //Устанавливем "флажок", показывая, что это число уже было использовано

}

}

printf("Сумма равна: %lf\n", SUM); //Выводим на экран искомую сумму

return 0;

}