Федеральное государственное образовательное бюджетное учреждение

высшего профессионального образования

«Нижегородский государственный университет им. Н.И. Лобачевского»

Институт информационных технологий, математики и механики

Отчёт по лабораторной работе

Работа с массивами и генерация случайных чисел

Выполнил:

студент института ИТММ

гр. 3821Б1ПМ3

Афанасьев К.О

Проверил:

Заведующий лабораторией Суперкомпьютерных технологий и высокопроизводительных вычислений

Лебедев И. Г.

Нижний Новгород

2021 г.

Содержание

[1. Введение 3](#_Toc86228936)

[2. Постановка задачи 4](#_Toc86228937)

[3. Руководство пользователя 5](#_Toc86228938)

[4. Руководство программиста 6](#_Toc86228939)

[1) Описание структуры программы 6](#_Toc86228940)

[2) Описание структур данных 7](#_Toc86228941)

[3) Описание алгоритмов 8](#_Toc86228942)

[5. Эксперименты 10](#_Toc86228943)

[6. Заключение 1](#_Toc86228944)1

[7. Литература](#_Toc86228945) 12

[8. Приложение 1](#_Toc86228946) 13

# Введение

Программирование можно рассматривать как искусство, науку, ремесло. Программирование - это искусство получения ответов от машины. Для этого в узком смысле нужно составить специальный код для технического устройства, а в широком - разработать программы на языках программирования, т.е. не просто составить код, а выполнить интеллектуальную работу по составлению высокоразумных программ для решения различных задач во всех сферах человеческой деятельности.

Программирование - процесс описания последовательности действий решения задачи средствами конкретного языка программирования и оформление результатов описания в виде программы. Эта работа требует точности, аккуратности и терпения. Команды машине должны формулироваться абсолютно четко и полно, не должны содержать никакой двусмысленности.

# Постановка задачи

Необходимо написать программу, которая будет генерировать множество случайных чисел размера N в диапазоне (min, max), где N, min, max вводятся с клавиатуры. После чего подсчитывает и выводит сумму, которая получается следующим образом: все числа, номера которых совпадают с дробной частью одного из исходных чисел - вычитаются, все остальные прибавляются.

# Руководство пользователя

1. Необходимо ввести данные, которые от тебя требует программа.

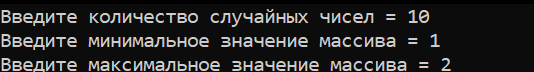


Рисунок 1. Ввод требуемых данных

1. Дальше пользователю нужно будет нажать Enter и компьютер сделает за него всё сам

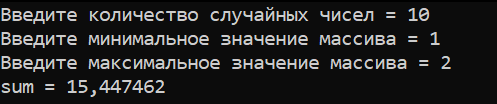


Рисунок 2. Вывод результата

# Руководство программиста

## Описание структуры программы

Условие для проверки данных

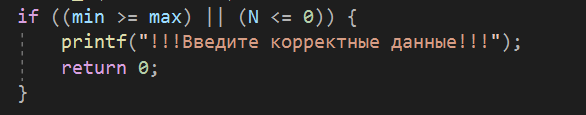


Рисунок 3.Проверка на корректность данных

Через цикл for происходит заполнение массива, подсчитывается сумма всех элементов и производится отделение дробной части при помощи функции modf. (рис. 4)

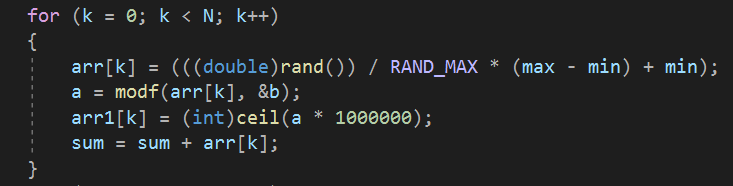


Рисунок 4. Цикл for для заполнения массива

Существует еще один вложенный цикл for для сравнения индексов массива с дробной частью числа (рис. 5)

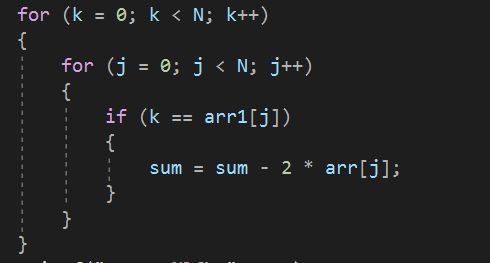


Рисунок 6. Цикл for для сравнения дробной части и индексов

Выводим сумму всех элементов (рис. 7)



Рисунок 7. Вывод результата

## Описание структур данных

В программе используются переменные разных типов и соответсвенно мы их объявляем (рис. 8)

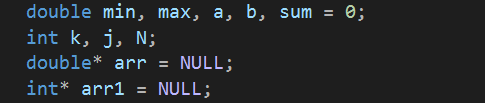


Рисунок 8. Объявление переменных

Также для работы используются следующие библиотеки (рис. 9)

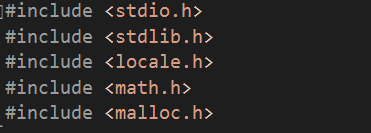


Рисунок 9. Подключение библиотеки

## Описание алгоритмов

Составим блок-схему, которая проверяет данные от пользователя на корректность.

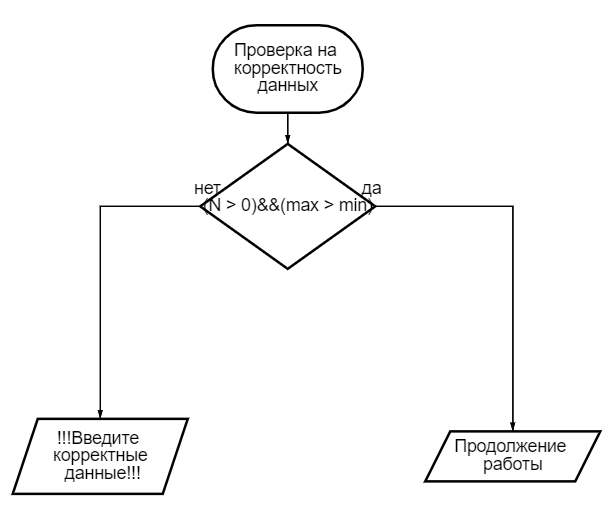


Рисунок 10. Блок схема проверки корректности данных

В modf подаётся число, которое нужно разделить на целую и дробную части. Modf отделяет их друг от друга и записывает значение целой части в переменную a, дробную в b

(рис. 11)



Рисунок 11. Функция modf

После того, как функция modf выделит дробную часть в переменную b, программа умножает полученное значение на 1.000.000 и при помощи функции ceil, которая округляет число к целому значению и приведения число к типу int, мы получаем число целого типа, которое записано в массив arr1 (рис. 12)



Рисунок12. Запись числа в arr1

Алгоритм for и его вложенный цикл сводится к тому, что k сравнивается со значением в arr1 и если они совпадают, то из суммы два раза вычитается значение arr1[j], потому что ранее мы добавили все элементы массива в sum (рис. 13)

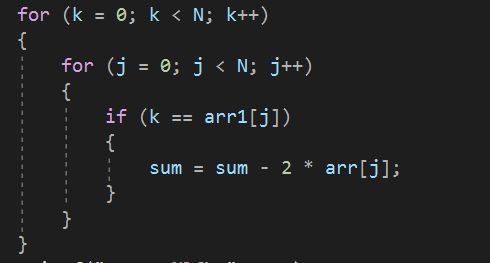


Рисунок 13. Цикл for и вложенный цикл для подсчета итогового результата

# Эксперименты

Проведем эксперименты. Возьмем N = 10 элементов, min = 1, max = 2 и посмотрим, что выдаст программа (рис. 14)

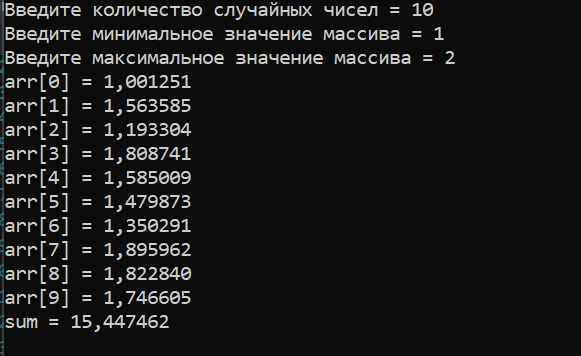


Рисунок 14. Работа программы при N =10, min = 1, max = 2

Заметим, что программа получила 10 случайных чисел в диапазоне от 1 до 2

Давайте проверим, работает ли проверка на корректный ввод данных.

(рис. 13) и (рис. 14)

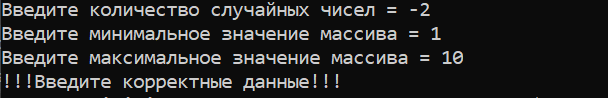


Рисунок 13. Ошибка при N <= 0

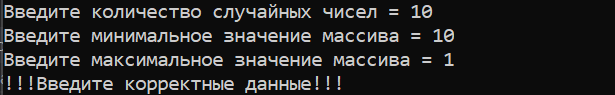


Рисунок 14. Ошибка при min > max

Видим, что проверка работает и выдает сообщение об ошибке.

# Заключение

В ходе данной лабораторной работы я создал программу, которая генерировала множество случайных чисел размера N в диапазоне (min, max), где N, min, max вводятся с клавиатуры. После чего подсчитывала и выводила сумму, которая получалась следующим образом: все числа, номера которых совпадали с дробной частью одного из исходных чисел - вычитались, все остальные прибавлялись. Я научился работать с массивами и с генерацией случайных чисел в заданном диапазоне. Я также подтянул свои знания в языке программирования С и улучшил практические навыки. В будущем я хотел бы и в дальнейшем изучать данный язык программирования и стать высокоуровневым специалистом в этой среде.

# Литература

* C. Полное руководство. Герберт Шилдт*

* Язык С в ХХI веке. Бен Клеменс*

* Изучаем программирование на C. Дэвид Гриффитс, Дон Гриффитс*

# Приложение 1

1. #include <stdio.h>
2. #include <stdlib.h>
3. #include <locale.h>
4. #include <math.h>
5. #include <malloc.h>
6. int main()
7. {
8. setlocale(LC\_ALL, "RUS");
9. double min, max, a, b, sum = 0;
10. int k, j, N;
11. double\* arr = NULL;
12. int\* arr1 = NULL;
13. printf("Введите количество случайных чисел = ");
14. scanf\_s("%d", &N);
15. printf("Введите минимальное значение массива = ");
16. scanf\_s("%lf", &min);
17. printf("Введите максимальное значение массива = ");
18. scanf\_s("%lf", &max);
19. if ((min >= max) || (N <= 0)) {
20. printf("!!!Введите корректные данные!!!");
21. return 0;
22. }
23. arr = ((double\*)malloc(N \* sizeof(double) + 1));
24. arr1 = ((int\*)malloc(N \* sizeof(double) + 1));
25. for (k = 0; k < N; k++)
26. {
27. arr[k] = (((double)rand()) / RAND\_MAX \* (max - min) + min);
28. //printf("arr[%d", k);
29. //printf("] = %lf\n", arr[k]);
30. a = modf(arr[k], &b);
31. arr1[k] = (int)ceil(a \* 1000000);
32. sum = sum + arr[k];
33. }
34. for (k = 0; k < N; k++)
35. {
36. for (j = 0; j < N; j++)
37. {
38. if (k == arr1[j])
39. {
40. sum = sum - 2 \* arr[j];
41. }
42. }
43. }
44. printf("sum = %lf\n", sum);
45. return 0;
46. }