Федеральное государственное образовательное бюджетное учреждение

высшего профессионального образования

«Нижегородский Государственный Университет им.

Н.И.Лобачевского» (ННГУ)

Институт Информационных Технологий Математики и Механики

Отчёт по лабораторной работе

Работа со случайно сгенерированными числами

Выполнил:

студент группы 3821Б1ФИ3

Загорулько А. А.

Проверил:

заведующий лабораторией суперкомпьютерных технологий и высокопроизводительных вычислений

Лебедев И. Г.

Нижний Новгород

2021г.

Содержание

[Введение 3](#_Toc85900684)

[Постановка задачи 4](#_Toc85900685)

[Руководство пользователя 5](#_Toc85900686)

[Руководство программиста 6](#_Toc85900687)

[Описание структуры кода программы 6](#_Toc85900688)

[Описание структуры данных 8](#_Toc85900689)

[Описание алгоритмов 9](#_Toc85900690)

[Эксперименты 10](#_Toc85900691)

[Заключение 11](#_Toc85900692)

[Список литературы 12](#_Toc85900693)

# Введение

Программирование — это процесс создания программ (программного обеспечения). Для этого программисты пишут исходный код на одном из языков программирования.

Случайные числа — это одна из основных составляющих любого языка программирования, на них строятся многие алгоритмы. Они имеют применение в физике, например, в исследованиях электронного шума, в инженерном деле и исследовании операций. Многие методы статистического анализа требуют использования случайных чисел.

В ходе выполнения лабораторный работы будет написана программа, работающая со случайными числами на языке «C»

# 

# Постановка задачи

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

## 

# Руководство пользователя

После запуска программа выводит сообщения “Input N”, “Input min”,” Input max”, после которых нужно ввести значения. (см. рис. 1)



Рисунок 1. Ввод данных от пользователя

После нажатии Enter, программа выведет результат суммирования сгенерированных чисел, произведенного по правилам технического задания. На этом программа завершается (см. рис. 2).

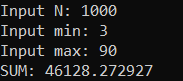


Рисунок 2. Результат работы программы

# 

# Руководство программиста

## Описание структуры программы

1. Подключение библиотек, с которыми предстоит работать.

|  |
| --- |
| #include <stdlib.h>  #include <math.h>  #include <stdio.h> |

1. Объявление переменных

|  |
| --- |
| int main(void)  int = 0, i, temp;  double min, max, fraction\_part\_number, a;  double num, sum = 0;  double\* mas;  int result = 0; |

1. Получение данных от пользователя и проверка их

|  |
| --- |
| printf("Input N: ");  scanf\_s("%d", &n);  printf("Input min: ");  scanf\_s("%lf", &min);  printf("Input max: ");  scanf\_s("%lf", &max);  if (min > max)  temp = min;  min = max;  max = temp; |

1. Заполнение массива mas рандомными числами в диапазоне от min до max.

|  |
| --- |
| for (int i = 0; i < n; i++)  {  mas[i] = (((double)rand()) / RAND\_MAX) \* (max - min) + min;  } |

## Отделяем дробную часть числа, посредством вычитания из исходного числа его целой копии и умножением этой разности на 10 в 6 степени. Проверяем, есть ли индекс в массиве, соответствующий дробной части числа. Если такой есть, то вычитаем элемент массива под этим индексом из переменной sum и зануляем его. Следующим циклом суммируем все элементы массива и значение переменной sum.

|  |
| --- |
| for (int i = 0; i < n; i++)  {  num = ((mas[i] - trunc(mas[i])) \* pow(10, 6));  result = (int)num;  while (result % 10 == 0) result = result / 10;  for (int j = 0; j < n; j++)  {  if (result == j)  {  sum = sum - mas[i];  mas[i] = 0;  }  }  }  for (int i = 0; i < n; i++)  sum = sum + mas[i]; |

1. Выводим значение переменной sum.

|  |
| --- |
| printf("SUM: %f\n\n", sum);  free(mas);  return 0; |

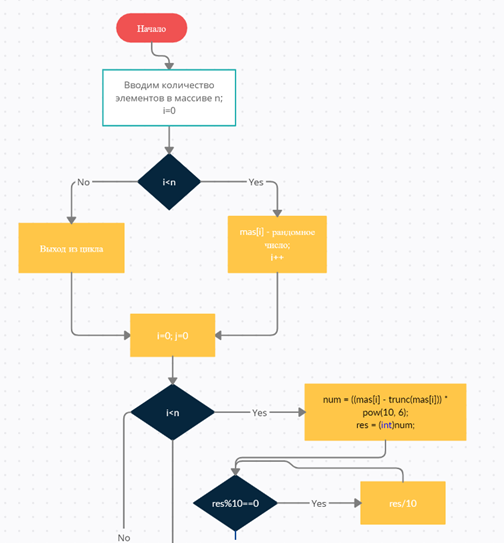
## Описание структуры данных

* В программе используются переменный счетчик типа int «i», количество генерируемых чисел «n» типа int.
* Переменные min, max и sum относятся к типу данных double.
* Случайные числа, создаваемые в ходе работы программы, записываются в массив типа double mas, а дробные части этих чисел записываютcя в переменную типа int res
* Функционирования программы требуются библиотеки: «stdio.h»; «stdlib.h» и «math.h».

## Описание алгоритмов

В программе реализованы два важных алгоритма:

1. Проверяет на подлинность вводимых данных. Имеется в виду, что, если минимальное число по ошибке пользователя будет больше максимального числа, тогда алгоритм произведет перестановку данных в этих переменных.
2. Выделение дробной части с фиксированной точностью и представление её как целого числа. Для этого из исходного числа вычитается его целая часть, полученное число умножается на 10 в степени 6 (количество знаков после запятой), приведение нового числа к типа int и присвоение переменной res значение полученного числа. А также обработка переменной res (деление на 10) при необходимости



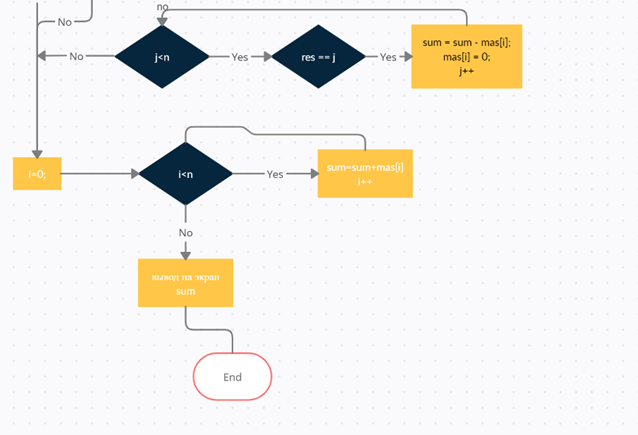


Рисунок 3. Блок-схема

**Эксперименты**

Написанную программу протестируем на различных входных данных, чтобы убедится в её работоспособности в различных условиях.

1. В случае ввода некорректных данных, таких как отрицательное количество элементов (см. рис.4)

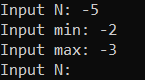


Рисунок 4.

1. В случае если вместо min ввести max, а вместо max ввести min (см. рис. 5)

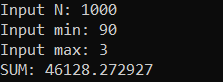


Рисунок 5.

**Заключение**

В ходе лабораторной работы мною была написана программа на языке «С», которая генерирует множество случайных чисел размера n в диапазоне (min, max), где n, min, max вводятся с клавиатуры. После чего подсчитывает, выводит сумму, которая получается следующим образом: все числа, номера которых совпадают с дробной частью одного из исходных чисел - вычитаются, все остальные прибавляются.

Выполнение данной лабораторной работы позволило мне понять принципы работы c циклами в языке «С», повысить уровень понимания при работе с различными алгоритмами, принципы работы со случайными числами, принципы работы с вводимыми пользователем данными их реализации внутри программы, их интерпретацию и проверку на корректность.

Выполнение подобных работ позволяет значительно улучшить свои навыки программирования, а также улучшить понимание работы с компьютером и средой разработки, в тоже время позволяет закрепить получаемые знания в области программирования и отработать их применения на практике.

**Список литературы**

1. Сборник задач по программированию/Д.М. Златопольский — 3-е изд., перераб. и доп. — СПб.: БХВ-Петербург, 2011. — 304 с.
2. C/C++. Программирование на языке высокого уровня / Т. А. Павловская. — СПб.: Питер, 2003. —461 с: ил.