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 №1

Работа с массивами случайно сгенерированных чисел

Выполнил:

студент группы 3821Б1ФИ3

Бендюжко Т.Г.

Проверил:

заведующий лабораторией суперкомпьютерных технологий и высокопроизводительных вычислений

Лебедев И.Г.

Нижний Новгород

2021г.

Содержание

[Введение 2](#_Toc85632832)

[Постановка задачи 3](#_Toc85632833)

[Руководство пользователя 4](#_Toc85632834)

[Руководство программиста 5](#_Toc85632835)

[Описание структуры кода программы 5](#_Toc85632836)

[Описание структуры данных 7](#_Toc85632837)

[Описание алгоритмов 7](#_Toc85632838)

[Эксперименты 10](#_Toc85632839)

[Заключение 13](#_Toc85632840)

Список литературы ……………………………………………………………………..14

Приложение 1……………………………………………………………………………15

Приложение 2……………………………………………………………………………16

# Введение

**Программирование** — раздел информатики, область знаний об алгоритмах и программах и их свойствах, а также исполнителях алгоритмов и программ.

Программирование включает в себя следующие разделы:

* Алгоритмика — наука об алгоритмах и исполнителях, наиболее сильно связанная с математикой. На стыке с математикой лежит [математическое программирование](http://www.wikiznanie.ru/wikipedia/index.php/%D0%9C%D0%B0%D1%82%D0%B5%D0%BC%D0%B0%D1%82%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%BE%D0%B5_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5).
* [Системное программирование](http://www.wikiznanie.ru/wikipedia/index.php?title=%D0%A1%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%BD%D0%BE%D0%B5_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5&action=edit&redlink=1) — дисциплина, связанная с написанием системного программного обеспечения. Основная сфера — собственно, программы, занимающиеся обработкой данных, и их представлением.
* [Прикладное программирование](http://www.wikiznanie.ru/wikipedia/index.php?title=%D0%9F%D1%80%D0%B8%D0%BA%D0%BB%D0%B0%D0%B4%D0%BD%D0%BE%D0%B5_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5&action=edit&redlink=1) — дисциплина, связанная с написанием прикладного, и отчасти инструментального программного обеспечения.

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 на языке программирования «С» будет написана программа, работающая со случайными числами, массивами и различными типами данных.

# 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 “Введите размер массива: ”, “Введите минимальную границу: ”, “Введите максимальную границу: ”, после которых нужно ввести соответствующие им значения.

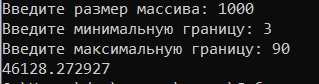


Рисунок 1. Ввод данных от пользователя

После нажатии Enter, программа выведет результат суммирования сгенерированных чисел, произведенного по правилам технического задания. На этом программа завершается (см. рис. 3).

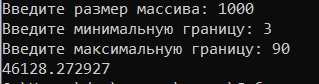


Рисунок 2. Результат работы программы

# Руководство программиста

## Описание структуры кода программы

1. Подключение библиотек, с которыми предстоит работать.

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

1. Объявление функции main, вначале которой мы объявляем переменные.

|  |
| --- |
| int main()  {  int max = 0;  int min = 0;  int n = 0;  int count = 0;  double num = 0;  int res = 0;  double sum = 0;  double\* mas; |

1. Получение данных от пользователя и проверка их на корректность.

|  |
| --- |
| do {  int ch=0;  printf("Введите размер массива: ");  count = scanf\_s("%d", &n);  while (ch = getchar() != '\n');  } while ((count < 1)||(n<0)||(n==0));  mas = (double\*)malloc(n \* sizeof(double));  do {  do {  int ch = 0;  printf("Введите минимальную границу: ");  count = scanf\_s("%d", &min);  while (ch = getchar() != '\n');  } while (count < 1);  do {  int ch = 0;  printf("Введите максимальную границу: ");  count = scanf\_s("%d", &max);  while (ch = getchar() != '\n');  } while (count < 1);  } while ((min > max)||(min==max)); |

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));  res = (int)num;  while (res % 10 == 0) res = res / 10;  for (int j = 0; j < n; j++) {  if (res == j) {  sum = sum - mas[i];  mas[i] = 0;  }  }  }  for (int i = 0; i < n; i++)  sum = sum + mas[i]; |

1. Печатаем значение переменной sum.

|  |
| --- |
| printf("\n%lf", sum);  return 0;  } |

## Описание структуры данных

* В программе используются переменные счетчики типа int i, j, count, количество генерируемых чисел n типа int.
* Переменные min, max относятся к типу int.
* Случайные числа, создаваемые в ходе работы программы, записываются в массив типа double mas, а дробные части этих чисел записываютcя в переменную типа int res.
* Результат программы записываются в переменную sum типа double.
* А также для функционирования программы требуются библиотеки: stdio.h, stdlib.h, math.h.

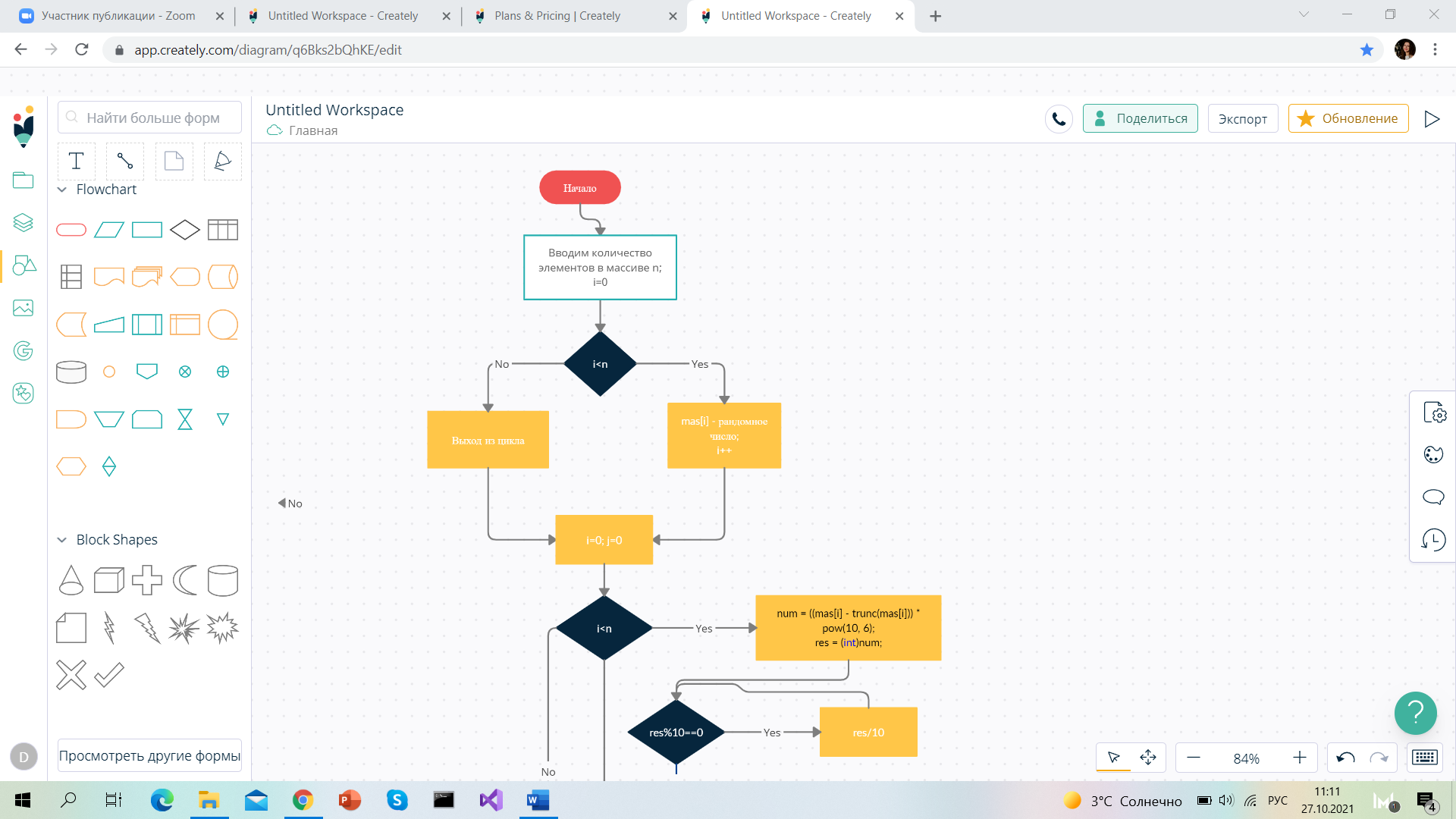
## Описание алгоритмов

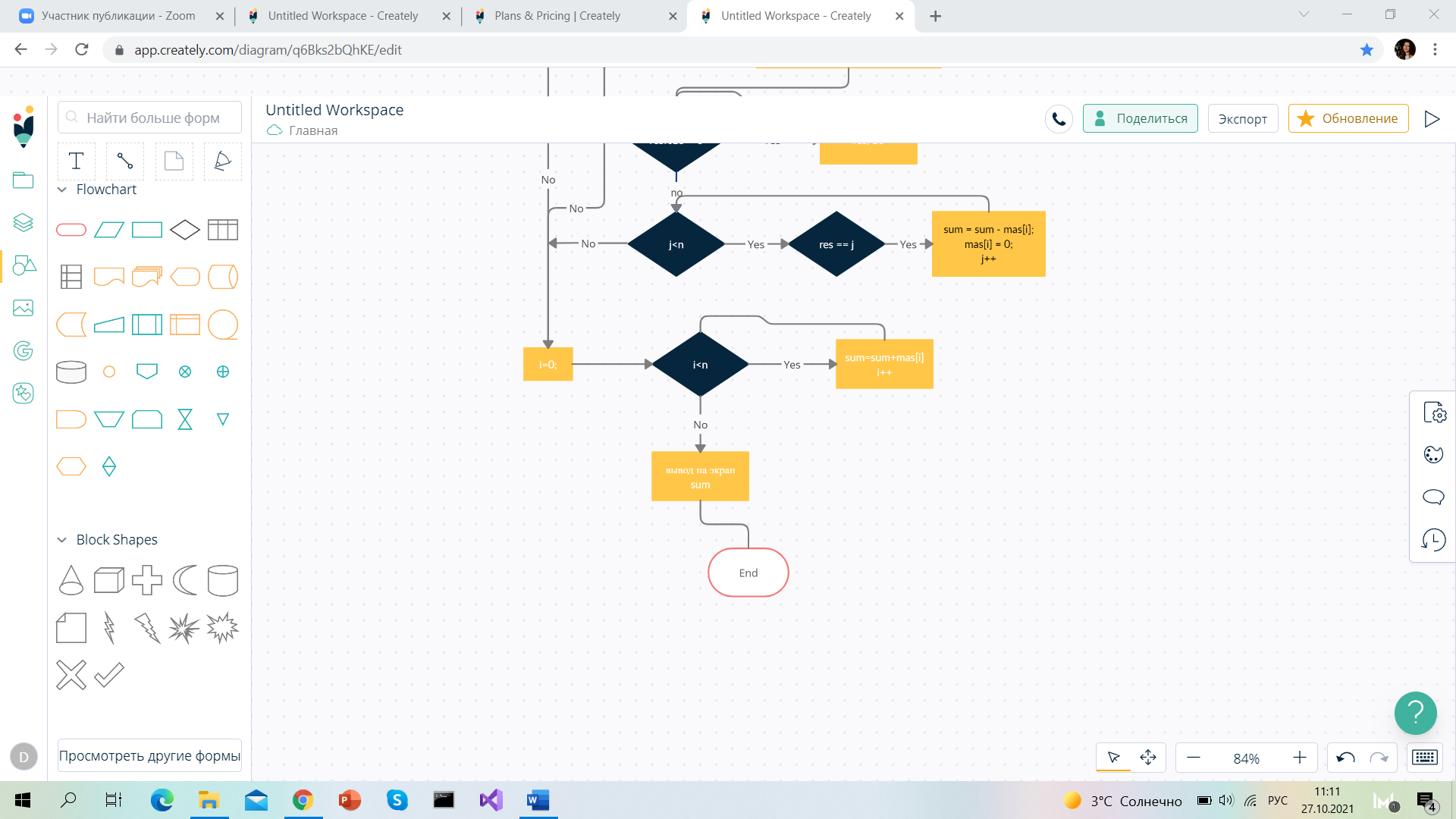
В данной программе реализовано несколько ключевых алгоритмов:

* 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.

|  |
| --- |
| num = ((mas[i] - trunc(mas[i])) \* pow(10, 6));  res = (int)num;  while (res % 10 == 0) res = res / 10; |

* Алгоритм заполнения массива случайными числами и подсчет суммы представлен следующей блок-схемой:





* Рисунок 3. Блок-схема

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

# 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 или неверный ввод границ диапазона, программа запросит ввод новых данных (см. рис. 4).

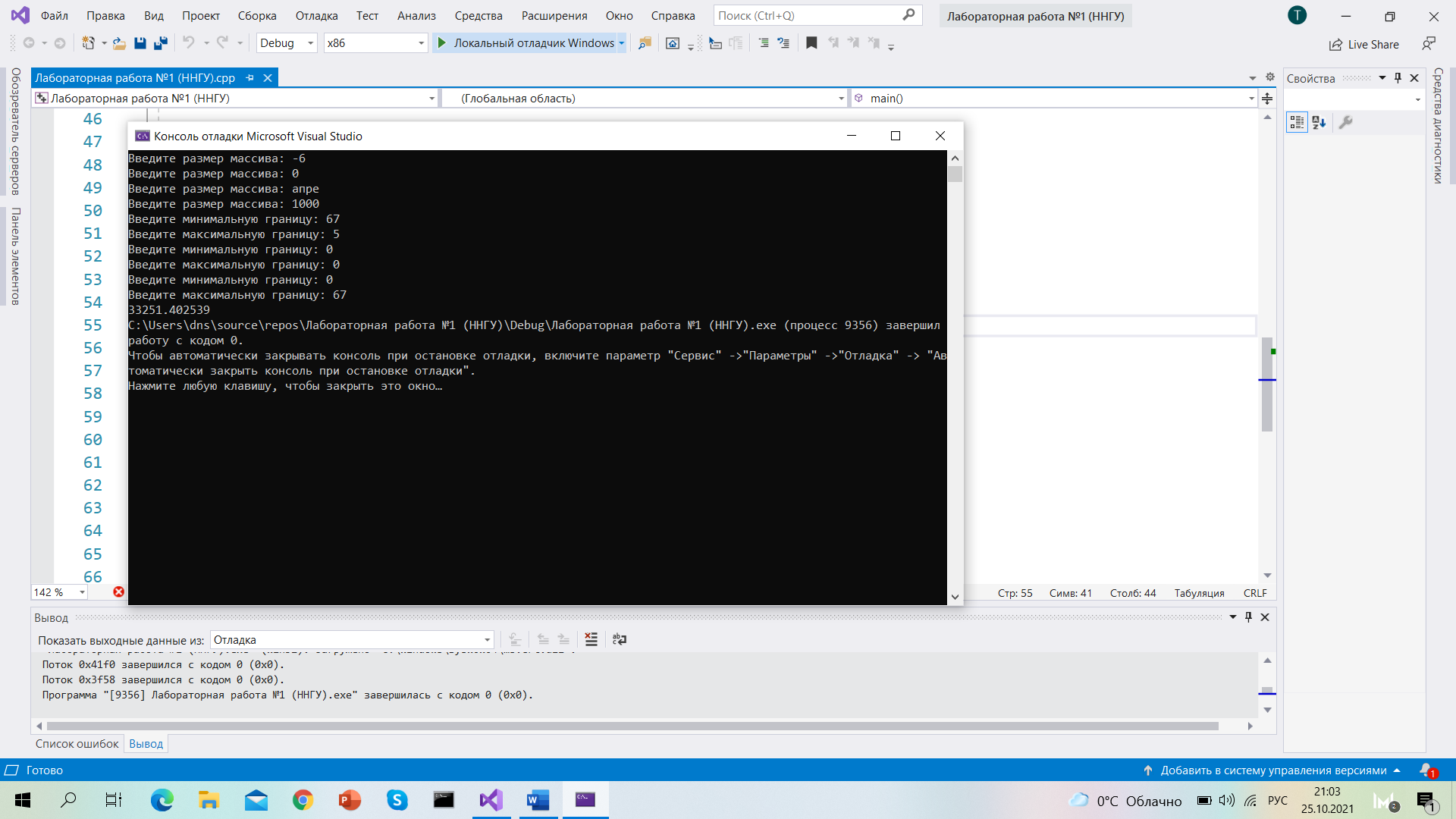


Рисунок 4. Ввод неверных данных

Теперь попробуем подать на вход различные корректные значения (см. рис. 6-7).

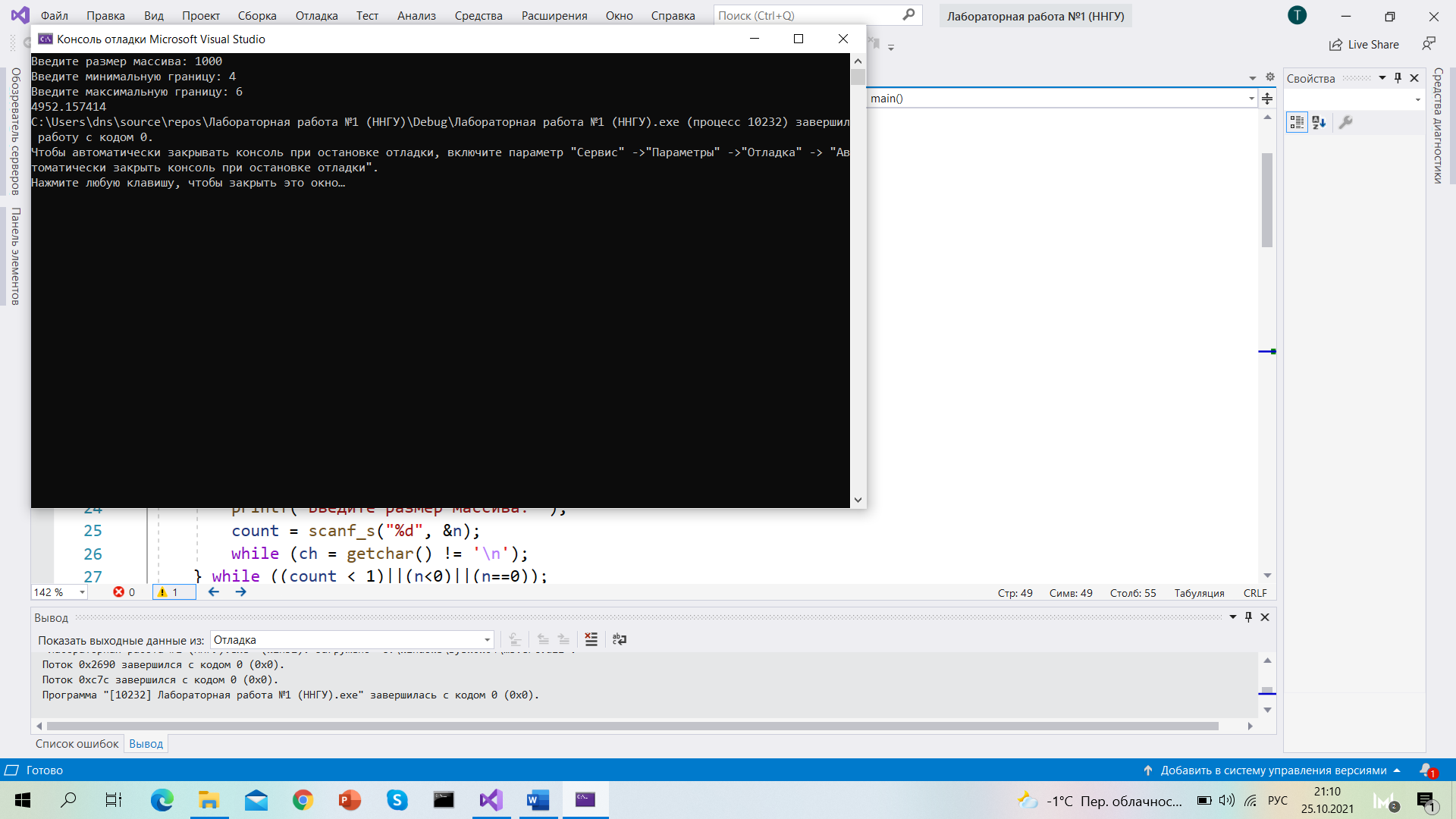


Рисунок 6. Работа программы при большом размере массива

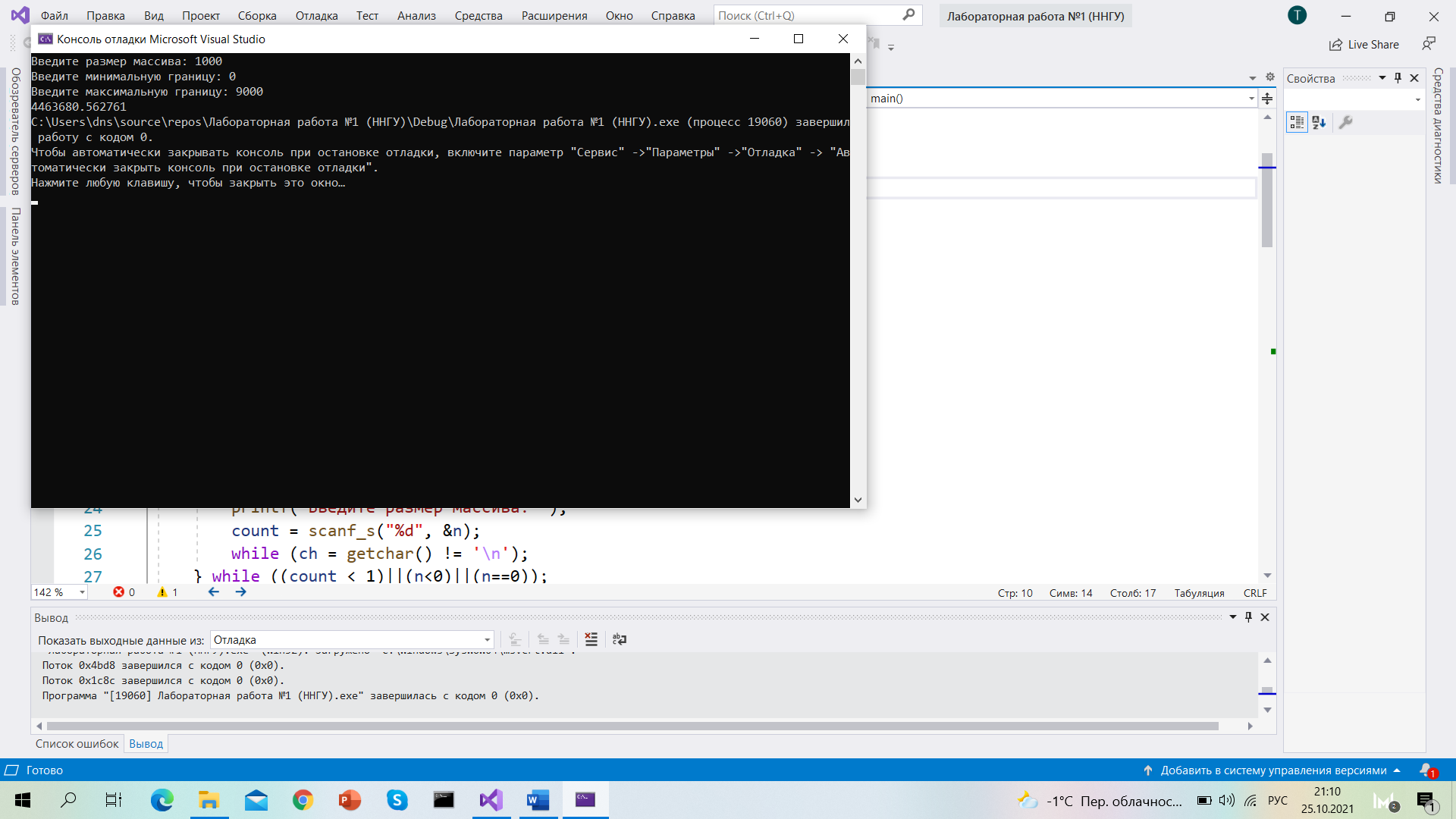


Рисунок 7. Работа программы при больших границах

При тестировании программы с различными входными данными в среде разработки Visual Studio никаких ошибок или предупреждений встречено не было.

# 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 лучше изучить принципы работы с динамическими массивами, изучить различные алгоритмы позволяющие работать с различными типами данных, данными в массивах, принципы работы со случайными числами, принципы работы с вводимыми пользователем данными их реализации внутри программы, их интерпретацию и проверку на корректность.

Список литературы

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

# Приложение 1

|  |
| --- |
| #include <iostream>  #include <stdlib.h>  #include <math.h>  int main()  {  system("chcp 1251");  system("cls");  int max = 0;  int min = 0;  int n = 0;  int count = 0;  double num = 0;  int res = 0;  double sum = 0;  double\* mas;  do {  int ch=0;  printf("Введите размер массива: ");  count = scanf\_s("%d", &n);  while (ch = getchar() != '\n');  } while ((count < 1)||(n<0)||(n==0));  mas = (double\*)malloc(n \* sizeof(double));  do {  do {  int ch = 0;  printf("Введите минимальную границу: ");  count = scanf\_s("%d", &min);  while (ch = getchar() != '\n');  } while (count < 1);  do {  int ch = 0;  printf("Введите максимальную границу: ");  count = scanf\_s("%d", &max);  while (ch = getchar() != '\n');  } while (count < 1);  } while ((min > max)||(min==max));    for (int i = 0; i < n; i++)  {  mas[i] = (((double)rand()) / RAND\_MAX) \* (max - min) + min;  }  for (int i = 0; i < n; i++) {  num = ((mas[i] - trunc(mas[i])) \* pow(10, 6));  res = (int)num;  while (res % 10 == 0) res = res / 10;  for (int j = 0; j < n; j++) {  if (res == j) {  sum = sum - mas[i];  mas[i] = 0;  }  }  }  for (int i = 0; i < n; i++)  sum = sum + mas[i];  printf\_s("%f", sum);  free(mas);  return 0;  } |

# Приложение 2

