ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«РОССИЙСКИЙ УНИВЕРСИТЕТ ТРАНСПОРТА»  
(РУТ (МИИТ))

Институт транспортной техники и систем управления

Кафедра «Управление и защита информации»

ОТЧЁТ  
О ПРАКТИЧЕСКОЙ РАБОТЕ № 3

По дисциплине «Языки программирования»

Выполнил: ст. гр. ТКИ – 141

Райхман И. А.

Проверил: к.т.н., доц.

Васильева М. А.

Москва 2022

Содержание

Задание 3–1 ............................................................................................ 3

Формулировка задания.......................................................................... 3

Блок-схема алгоритма ........................................................................... 3

Программа на языке С++ ...................................................................... 4

Решение тестовых примеров на С++.................................................... 6

Решение тестовых примеров в Excel.................................................... 7

Зачет задания в GitHub...........................................................................10

Задание 3–2 ............................................................................................. 11

Формулировка задания......................................................................... 11

Блок-схема алгоритма .......................................................................... 11

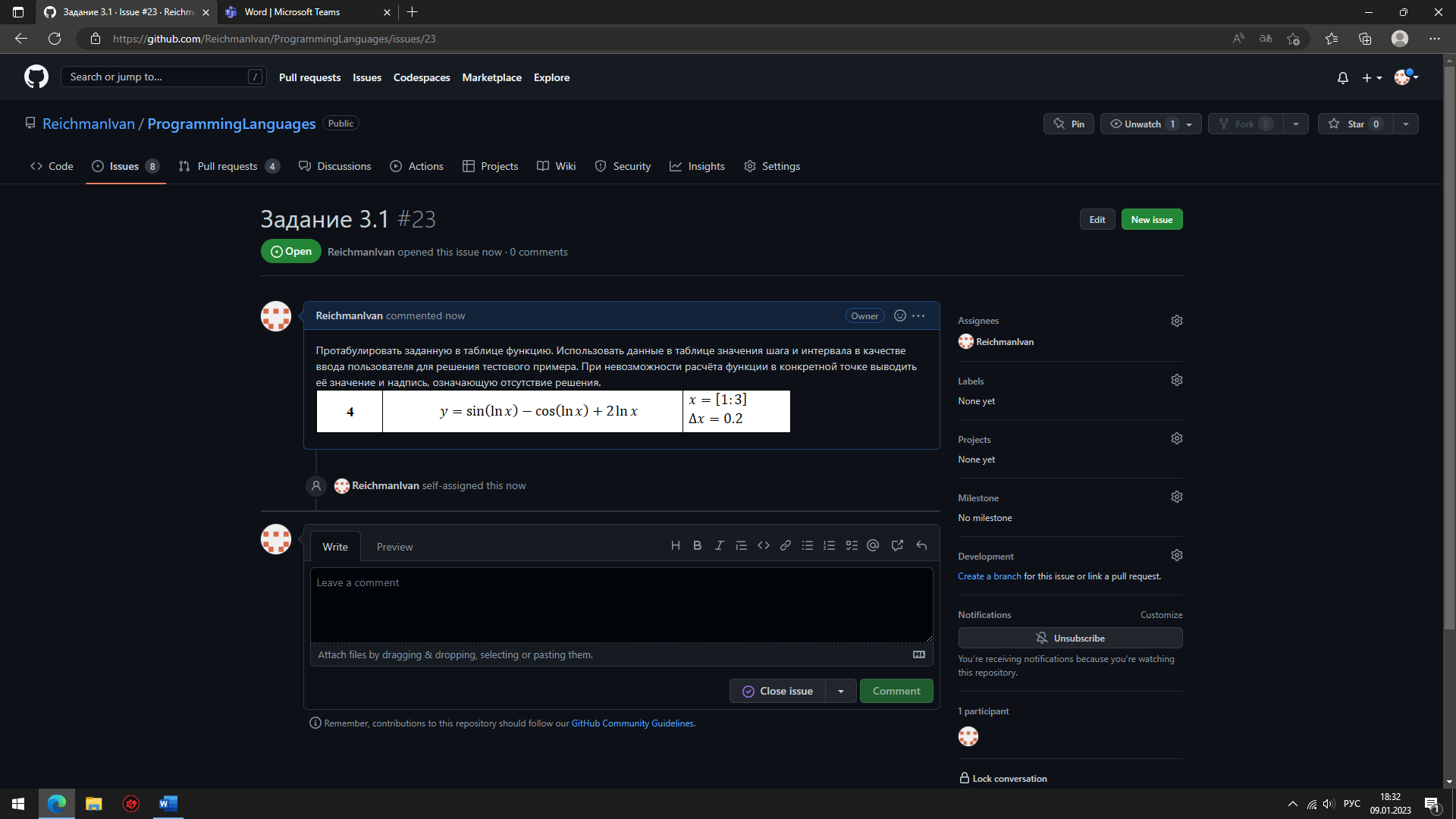
Программа на языке С++ ................................................................... 13

Решение тестовых примеров на С++.................................................. 15

Зачет задания в GitHub...........................................................................16

# Задание 3–1

1. Формулировка задачи



1. Блок-схема алгоритма

Блок-схемы алгоритмов функций представлены на рисунках (Рисунок 1, Рисунок 2, Рисунок 3).

Изображение выглядит как текст, компьютер, дисплей

Автоматически созданное описание

Рисунок 1 – Блок-схема функции main()

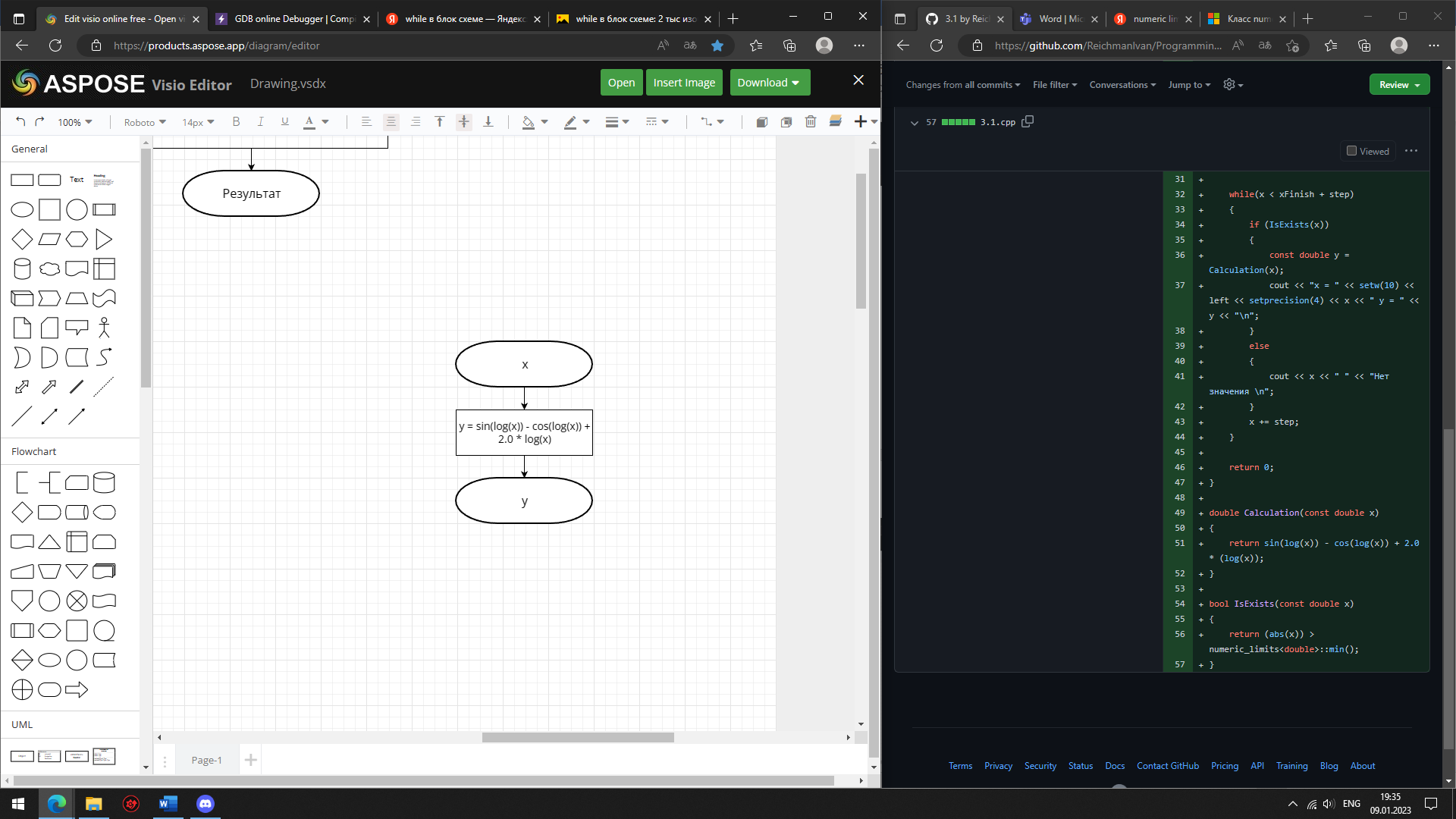


Рисунок 2 – Блок-схема функции Calculation(x)

Изображение выглядит как текст, снимок экрана, дисплей

Автоматически созданное описание

Рисунок 3 – Блок-схема функции IsExists(x)

1. Решение задачи на языке программирования C++

#include <iostream>

#include <cmath>

#include <iomanip>

#include <limits>

using namespace std;

/\*\*

\* \brief Функция расчета по заданной формуле.

\* \param x Параметр функции.

\* \return Возвращает значение функции.

\*/

double Calculation(const double x);

/\*\*

\* \brief Проверка y при заданном x.

\* \param x Параметр функции.

\* \return true Если y существует при заданном x.

\*/

bool IsExists(const double x);

/\*\*

\* \brief Точка входа в программу.

\* \return Возвращает 0 в случае успешного выполнения.

\*/

int main()

{

const double xStart = 1.0;

const double xFinish = 3.0;

const double step = 0.2;

double x = xStart;

while(x < xFinish + step)

{

if (IsExists(x))

{

const double y = Calculation(x);

cout << "x = " << setw(10) << left << setprecision(4) << x << " y = " << y << "\n";

}

else

{

cout << x << " " << "Нет значения \n";

}

x += step;

}

return 0;

}

double Calculation(const double x)

{

return sin(log(x)) - cos(log(x)) + 2.0 \* (log(x));

}

bool IsExists(const double x)

{

return (abs(x)) > numeric\_limits<double>::min();

}

1. Решение тестовых примеров

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 4 – Решение тестовых примеров

1. Решение тестовых примеров в Excel

Изображение выглядит как стол

Автоматически созданное описаниеИзображение выглядит как стол

Автоматически созданное описаниеИзображение выглядит как стол

Автоматически созданное описаниеИзображение выглядит как стол

Автоматически созданное описаниеИзображение выглядит как стол

Автоматически созданное описаниеИзображение выглядит как стол

Автоматически созданное описаниеИзображение выглядит как стол

Автоматически созданное описаниеИзображение выглядит как стол

Автоматически созданное описаниеИзображение выглядит как стол

Автоматически созданное описаниеИзображение выглядит как стол

Автоматически созданное описаниеИзображение выглядит как стол

Автоматически созданное описание

Рисунок 5 – Решение тестовых примеров

1. Зачет задания в GitHub

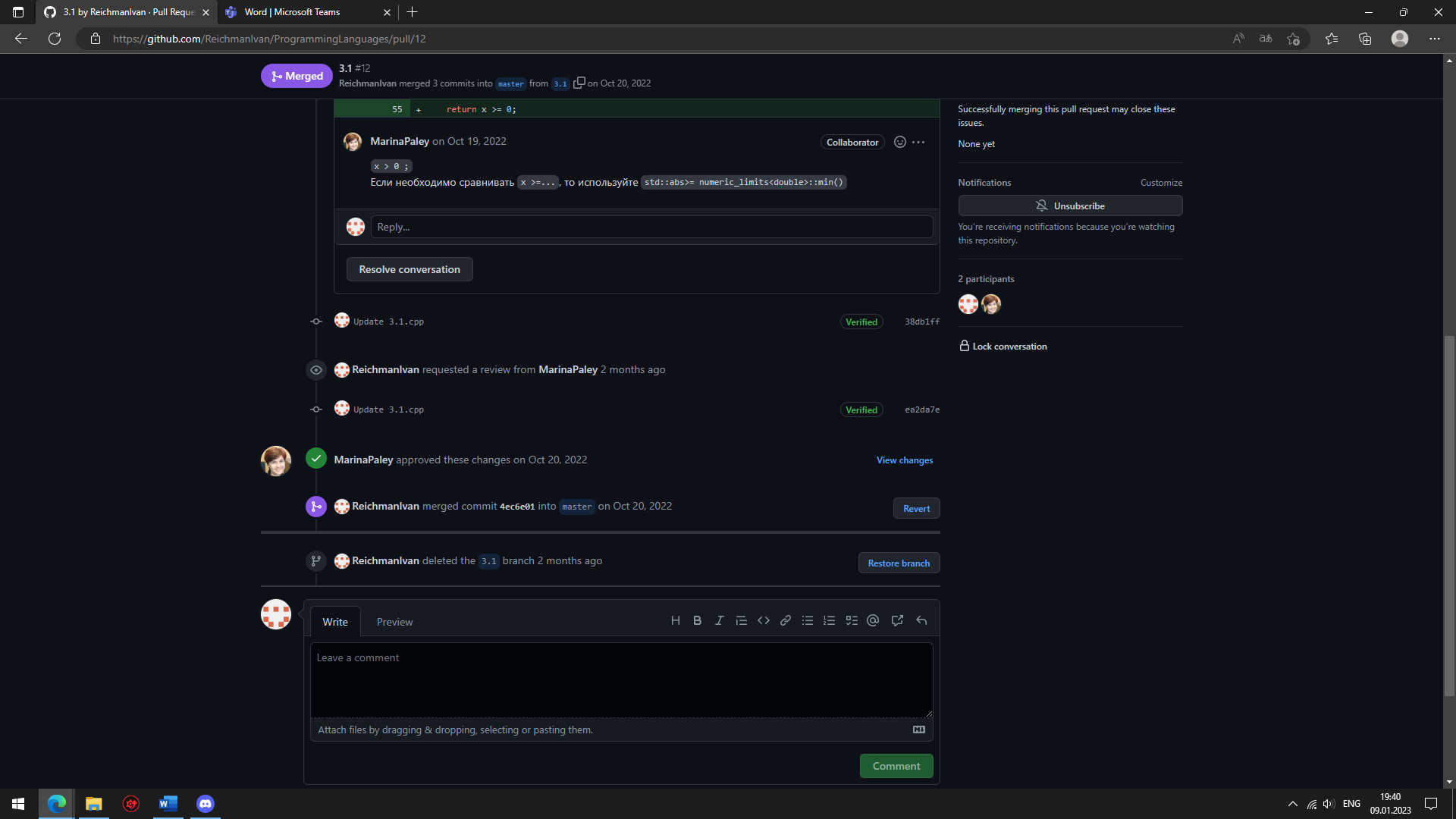


Рисунок 6 – Зачет задания

# Задание 3–2

1. Формулировка задачи

Изображение выглядит как текст, монитор, снимок экрана, черный

Автоматически созданное описание

1. Блок-схема алгоритма

Блок-схемы алгоритмов функций представлены на рисунках (Рисунок 1, Рисунок 2, Рисунок 3).

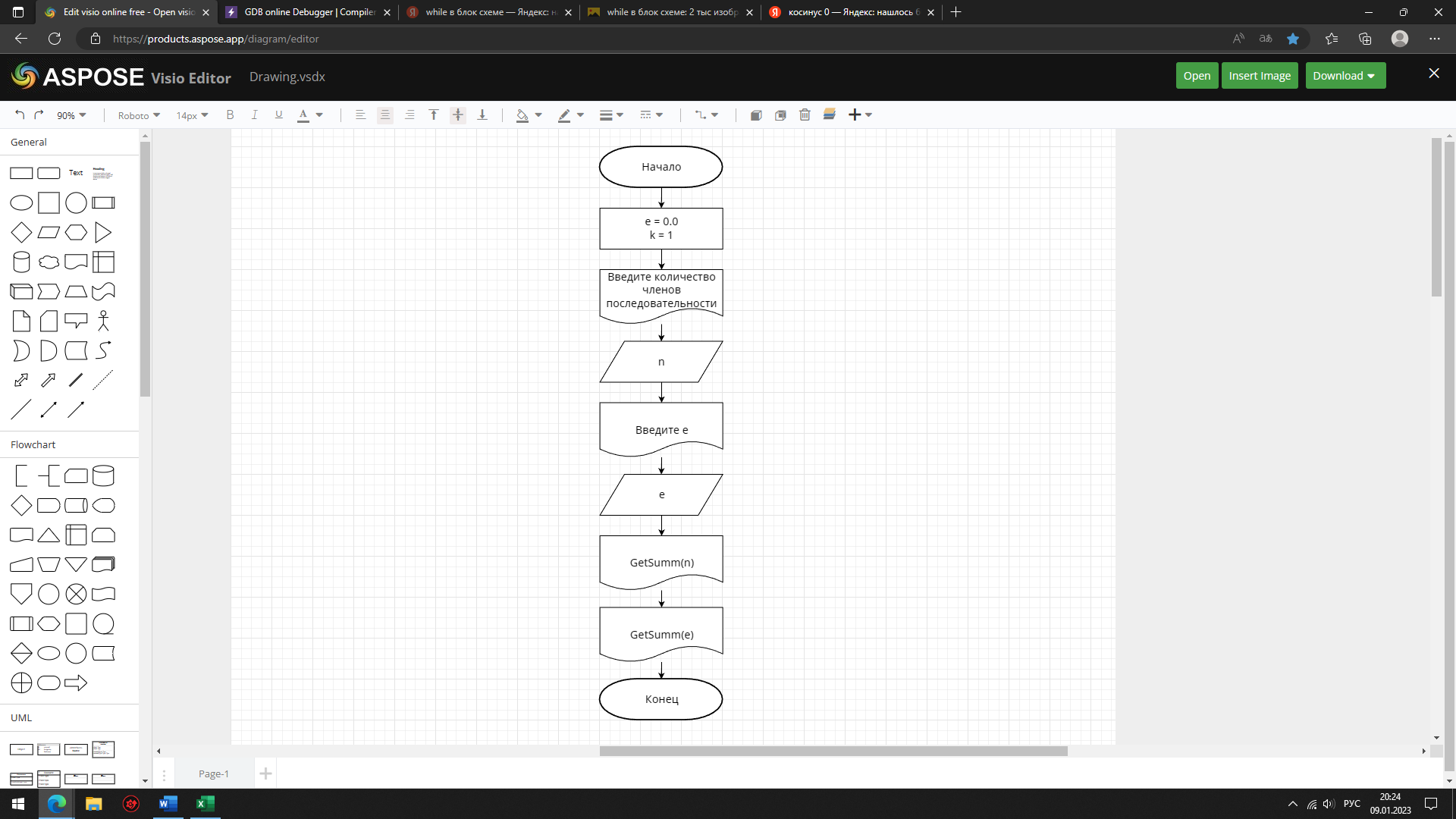


Рисунок 1 – Блок-схема функции main()

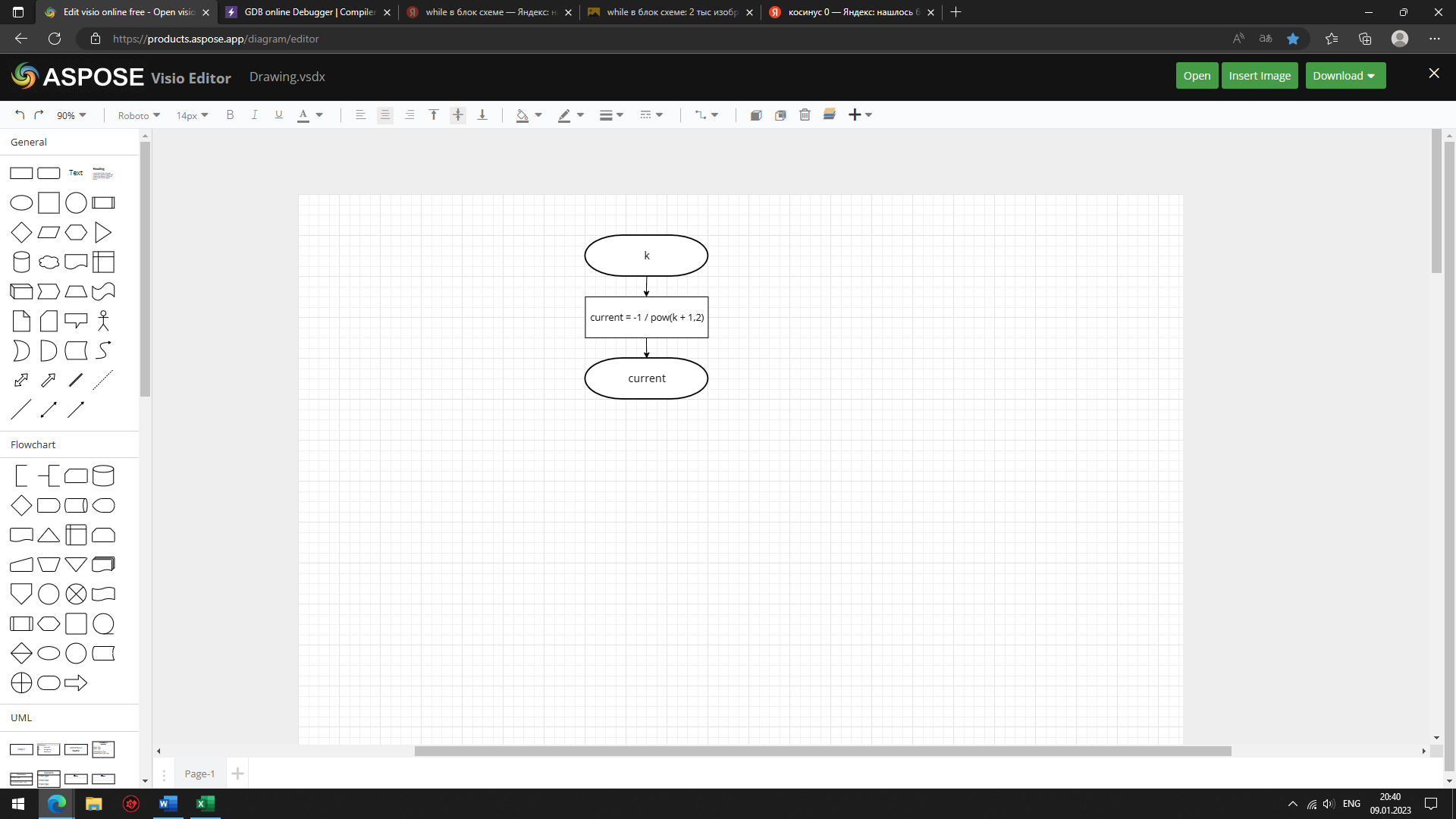


Рисунок 2 – Блок-схема функции GetCurrent(k)

Изображение выглядит как текст, компьютер, внутренний, электроника

Автоматически созданное описание

Рисунок 3 – Блок-схема функции GetSumm(n)

Изображение выглядит как текст, компьютер, внутренний, дисплей

Автоматически созданное описание

Рисунок 4 – Блок-схема функции GetSumm(e)

1. Решение задачи на языке программирования C++

#include <iostream>

#include <cmath>

using namespace std;

/\*

\*\brief функция определения текущего элемента ряда.

\*\param k - переменная для вычисления элемента ряда.

\*\return значение текущего элемент ряда.

\*/

double GetCurrent(const size\_t k);

/\*

\* \brief функция вычисления суммы первых n элементов ряда.

\* \param n - количество элементов ряда.

\* \return сумма первых n элементов ряда.

\*/

double GetSumm(const int n);

/\*

\* \brief функция вычисления суммы членов ряда не меньших числа e.

\* \param e - константа по условию.

\* \return сумма членов ряда не меньших числа e.

\*/

double GetSumm(const double e);

/\*

\* \brief точка входа в программу.

\* \return возвращает 0 в случае успеха.

\*/

int main()

{

int n;

double e = 0.0;

size\_t k = 1;

cout << "Введите количество членов последовательности" << endl;

cin >> n;

cout << "Введите e" << endl;

cin >> e;

cout << GetSumm(n) << endl;

cout << GetSumm(e);

return 0;

}

double GetCurrent(const size\_t k)

{

return -1 / pow(k + 1,2);

}

double GetSumm(const int n)

{

double sum = 0.0;

for (size\_t k = 1; k <= n; k++)

{

sum = sum + GetCurrent(k);

}

return sum;

}

double GetSumm(const double e)

{

int k = 1;

double sum = 0.0;

while (abs(GetCurrent(k)) >= e)

{

sum = sum + GetCurrent(k);

k += 1;

}

return sum;

}

1. Решение тестовых примеров

Изображение выглядит как текст, монитор, снимок экрана

Автоматически созданное описаниеИзображение выглядит как текст

Автоматически созданное описание

Рисунок 4 – Решение тестовых примеров при е = 1

Изображение выглядит как текст

Автоматически созданное описание Изображение выглядит как текст, снимок экрана, монитор

Автоматически созданное описание

Рисунок 5 – Решение тестовых примеров при е = 0.01

1. Зачет задания в GitHub

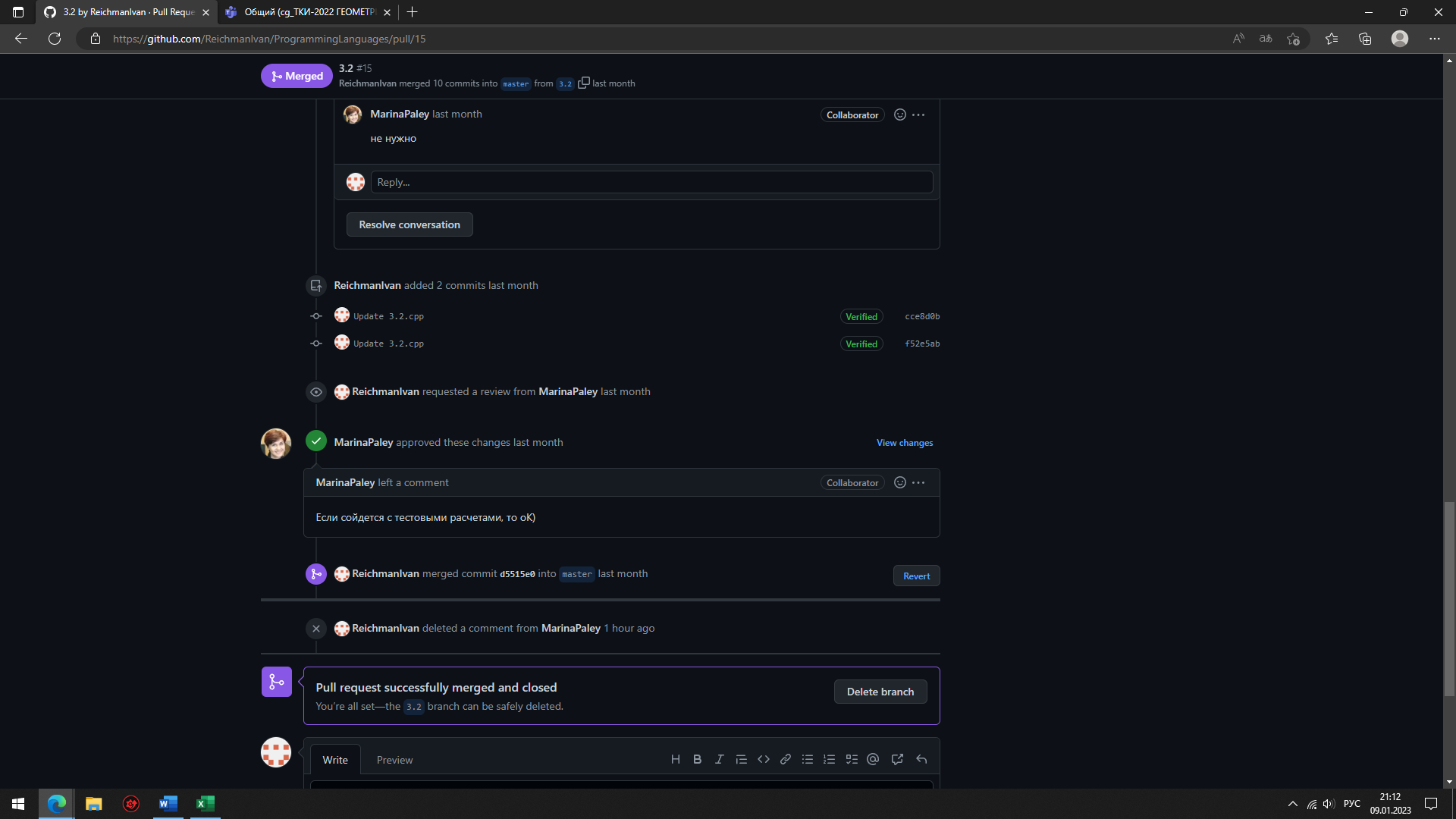


Рисунок 8 – Зачет задания