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

Институт транспортной техники и систем управления

Кафедра «Управление и защита информации»

ОТЧЁТ  
ПО ЛАБОРАТОРНОЙ РАБОТЕ №3

По дисциплине «Языки программирования»

ВАРИАНТ 12

Выполнил: ст. гр. ТКИ-141

Комаров Константин Сергеевич

Проверил: к.т.н., доц. Васильева М. А.

(Проверил: к.т.н, доц. Балакина Е. П.)

Москва 2024

**РЕШЕНИЕ ЗАДАЧИ 3-1**

* 1. **Формулировка задачи 3-1**

Протабулировать заданную в таблице функцию. Использовать данные в таблице значения шага и интервала в качестве ввода пользователя для решения тестового примера. При невозможности расчёта функции в конкретной точке выводить её значение и надпись, означающую отсутствие решения.

Таблица  – Исходные данные

****

* 1. **Блок-схема алгоритма задания 3-1**

Блок-схема основного алгоритма представлена ниже (Рисунок 1). Блок – схемы функций ***CheckInterval*** и ***ChekStep*** представлены ниже (Рисунок 2). Блок - схемы проверки ввода (Рисунок 3). Блок – схема функции ***GetFunctionValue*** (Рисунок 4).

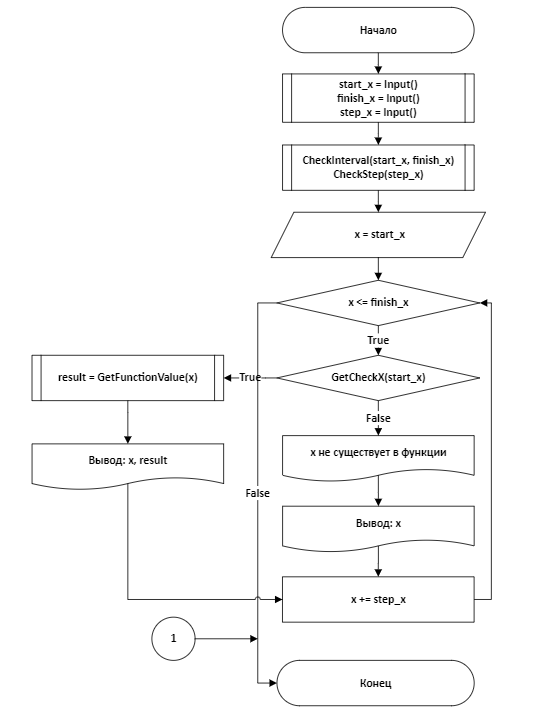


Рисунок – Блок-схема основного алгоритма

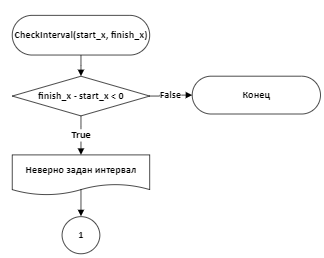
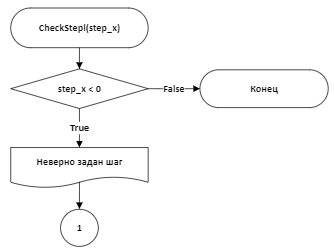
 

Рисунок 2 – Блок-схема функций ***CheckInterval*** и ***ChekStep***

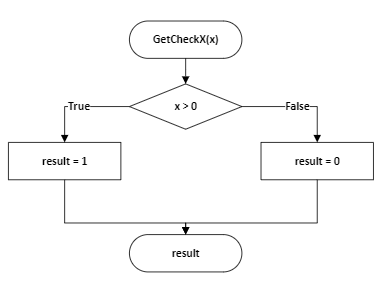
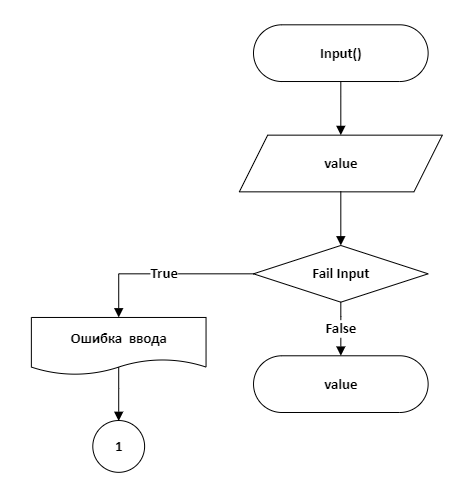


Рисунок 3 – Блок-схема функций проверки ввода

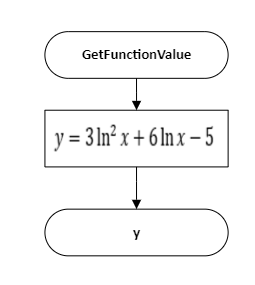


Рисунок 4 – Блок-схема функции ***GetFunctionValue***

* 1. **Текст программы на языке программирования С задание 3-1**

#include <stdio.h>

#include <math.h>

#include <stdlib.h>

#include <float.h>

#include <locale.h>

#include <stdbool.h>

/\*

\* @brief Считает нашу функцию.

\* @param x - значение параметра.

\* @return Значение функции.

\*/

double GetFunctionValue(const double x);

/\*

\* @brief Проверяет, можно ли вычислить функцию для заданного значения x.

\* @param x Значение x, которое требуется проверить.

\* @return true, если функцию можно вычислить для данного значения x, false в противном случае.

\*/

bool GetCheckX(const double x);

/\*

\* @brief Функция проверки ввода.

\* @return Возвращает значение, если верное, иначе выдает ошибку.

\*/

double Input(void);

/\*

\* @brief Проверяет коректность заданного интервала.

\* @param start\_x начальное значение интервала.

\* @param finish\_x конечное значение интервала.

\* @return Возвращает ошибку если интервал задан неверно.

\*/

void CheckInterval(const double start\_x, const double finish\_x);

/\*

\* @brief Проверяет коректность заданного шага.

\* @param step\_x значение шага.

\* @return Возвращает ошибку если шаг задан неверно.

\*/

void CheckStep(const double step\_x);

/\*

\* @brief Точка входа в программу.

\* @return Возвращает значение функции с заданным в цикле корнем.

\*/

int main(void)

{

setlocale(LC\_ALL, "Russian");

puts("Введите начальное значение x:");

const double start\_x = Input();

puts("Введите конечное значение x:");

const double finish\_x = Input();

puts("Введите значение шага");

const double step\_x = Input();

CheckInterval(start\_x, finish\_x);

CheckStep(step\_x);

double x = start\_x;

while (x <= finish\_x + DBL\_EPSILON)

{

if (GetCheckX(start\_x))

{

printf("x = %lf y = % lf\n", x, GetFunctionValue(x));

}

else

{

printf("X не существет в функции %2lf\n", x);

}

x += step\_x;

}

return 0;

}

bool GetCheckX(const double x)

{

return x > -DBL\_EPSILON;

}

void CheckInterval(const double start\_x, const double finish\_x)

{

if (finish\_x - start\_x < DBL\_EPSILON)

{

puts("Неверно задан интервал");

exit(EXIT\_FAILURE);

}

}

void CheckStep(const double step\_x)

{

if (step\_x <= DBL\_EPSILON)

{

puts("Неверно задан шаг");

exit(EXIT\_FAILURE);

}

}

double Input(void)

{

double value = 0.0;

if (scanf\_s("%lf", &value) != 1)

{

puts("Ошибка Input");

exit(EXIT\_FAILURE);

}

return value;

}

double GetFunctionValue(const double x)

{

return 3 \* pow(log(x), 2) + 6 \* log(x) - 5;

}

**1.4 Результаты выполнения программы 3-1**

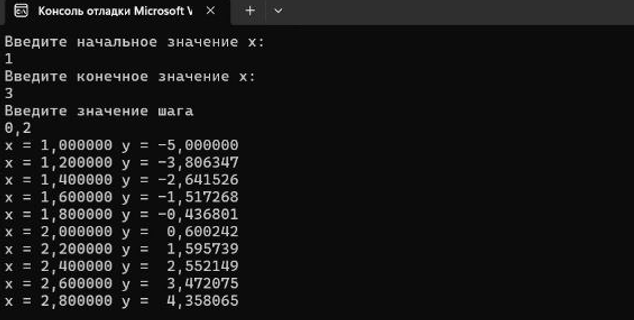


Рисунок 5 – Вывод значений функции в заданных точках

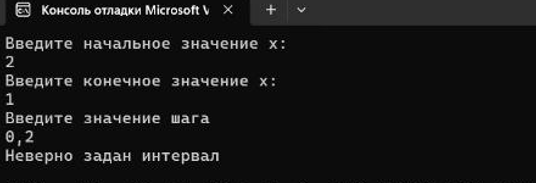


Рисунок 6 – Вывод ошибки при неправильно заданом интервале

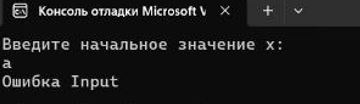


Рисунок 7 – Вывод ошибки ввода

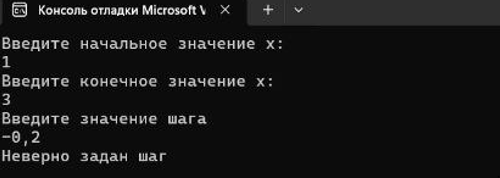


Рисунок 8 – Вывод при неправильном заданном шаге

**1.5 Выполнение тестовых примеров задание 3-1**

В программе MC Excelвыполнены тестовые примеры. Результаты их выполнения представлены ниже (Рисунок 9).

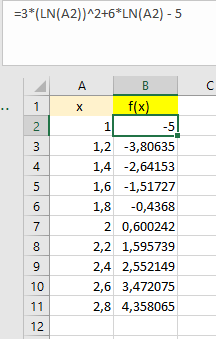
****

Рисунок 9 – Тестовое решение

**1****.6 Отметка о выполнении задания в веб-хостинге системы контроля версий задание 3-1**

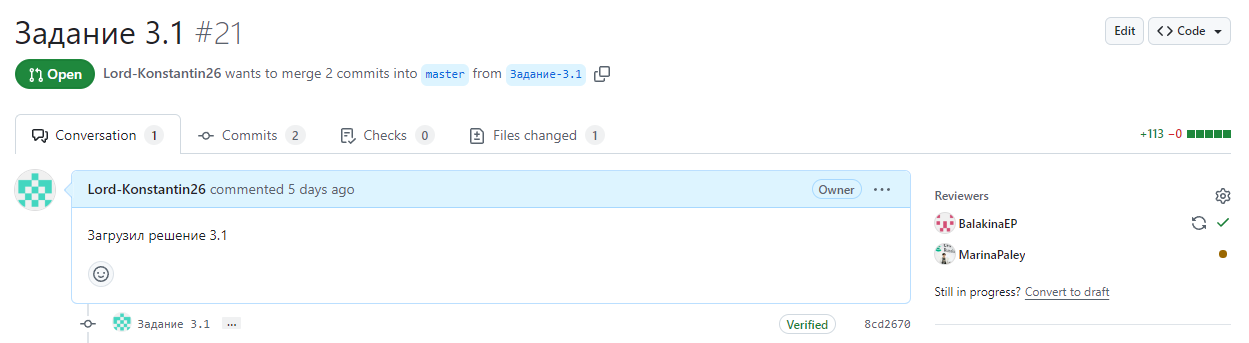
Ниже представлено доказательство того, что задание 3–1 было принято. (10)

Рисунок 10 – Approve задачи 3–1