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 ЛАБОРАТОРНОЙ РАБОТЕ №3

По дисциплине «Языки программирования»

ВАРИАНТ 6

Выполнил: ст. гр. ТКИ-142

Дохолян Герман Оганесович

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

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

Москва 2024

1. Формулировка задания

Протабулировать заданную в таблице функцию (Таблица 1). 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 1 – Исходные данные

|  |  |  |
| --- | --- | --- |
| Вариант | Функция | Константы |
| 6 |  |  |

1. Блок-схема алгоритма

Блок-схема основного алгоритма представлена ниже (Рисунок 1). Блок-схемы функций проверки величин, расчета функции и ввода представлены ниже (Рисунок 2, 3, 4, 5, 6).

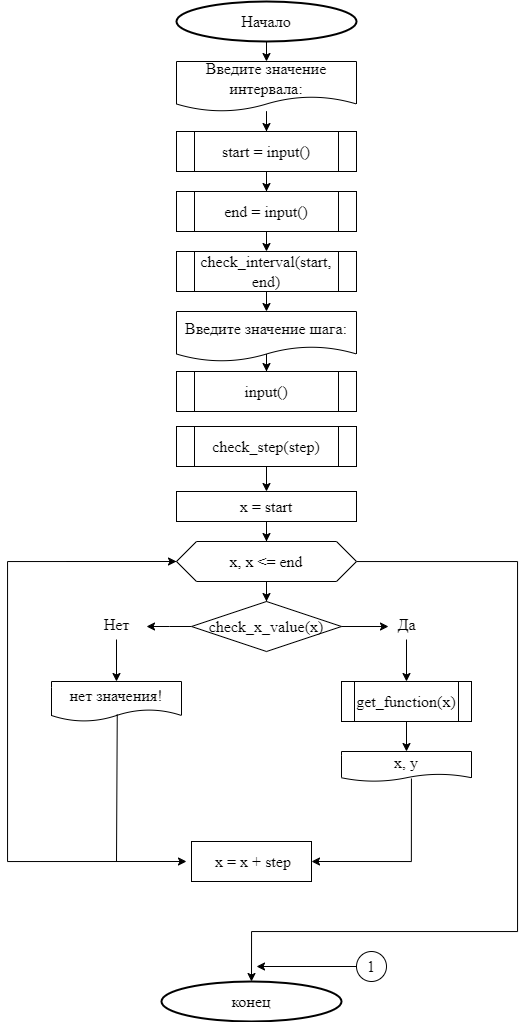


Рисунок 1 ­ Блок-схема основного алгоритма

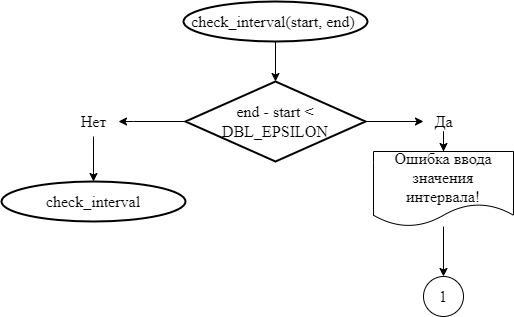


Рисунок 2 – Блок-схема функции check\_interval(start, end)

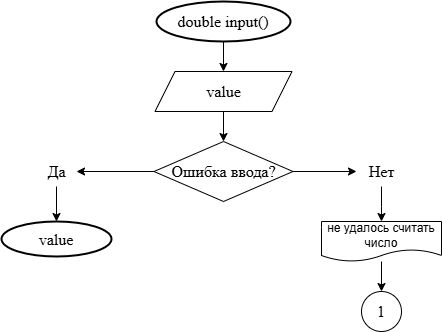


Рисунок 3 – Блок-схема функции input()

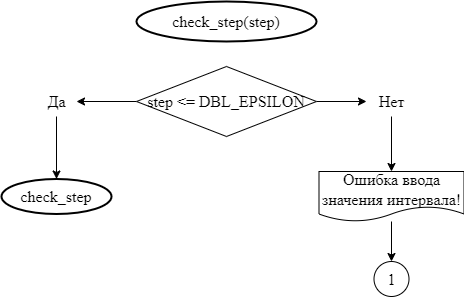


Рисунок 4 – Блок-схема функции check\_step(step)

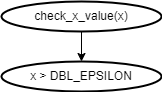


Рисунок 5 – Блок-схема функции check\_ x\_value (x)

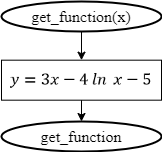


Рисунок 6 – Блок-схема функции get\_function(x)

1. Текст программы на языке C

#include <stdio.h>

#include <stdlib.h>

#include <errno.h>

#include <stdbool.h>

#include <float.h>

#include <math.h>

/\*\*

\* @brief Считывает вещественное значение

\* @return Возвращает вещественное значение

\*/

double input(void);

/\*\*

\* @brief Проверяет правильность введенных значений интервала

\* @param start начало интервала

\* @param end конец интервала

\* @return Возвращает ошибку, если введен некорректный интервал

\*/

void check\_interval(const double start, const double end);

/\*\*

\* @brief Проверяет правильность введенного значения шага

\* @param step величина шага

\* @return Возвращает ошибку, если введен некорректный шаг

\*/

void check\_step(const double step);

/\*\*

\* @brief Проверят правильность значения x

\* @param x переменная x

\* @return Возвращает True в случае успеха, иначе - False

\*/

bool check\_x\_value(double x);

/\*\*

\* @brief Вызвает функцию

\* @param x значение переменной x

\* @return Возвращает значение функции если нет деления на нуль, в ином случае вызывает ошибку

\*/

double get\_function(double x);

/\*\*

\* @brief Точка входа в программу

\* @return Возвращает 0 в случае успеха

\*/

int main(void)

{

printf("Введите значение интервала: ");

const double start = input();

const double end = input();

check\_interval(start, end);

printf("Введите значение шага: ");

const double step = input();

check\_step(step);

for (double x = start; x < end + step; x += step)

{

if (!check\_x\_value(x))

{

printf("нет значения!");

}

else

{

printf("x = %.3lf, y = %.3lf\n", x, get\_function(x));

}

}

return 0;

}

double input()

{

double value = 0.0;

int result = scanf("%lf", &value);

if (result != 1)

{

errno = EIO;

perror("Ошибка ввода!");

exit(EXIT\_FAILURE);

}

return value;

}

void check\_interval(const double start, const double end)

{

if (end - start < DBL\_EPSILON)

{

errno = EIO;

perror("Ошибка ввода значения интервала!");

exit(EXIT\_FAILURE);

}

}

void check\_step(const double step)

{

if (step <= DBL\_EPSILON)

{

errno = EIO;

perror("Ошибка ввода значения шага!");

exit(EXIT\_FAILURE);

}

}

bool check\_x\_value(double x)

{

return x > DBL\_EPSILON;

}

double get\_function(double x)

{

return 3 \* x - 4 \* log(x) - 5;

}

1. Результаты выполнения программы

Результаты выполнения программы представлены ниже (Рисунок 6, 7, 8, 9, 10, 11).

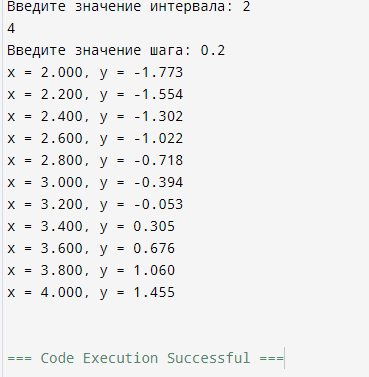


Рисунок 7 – Результаты табулирования функции с исходными данными

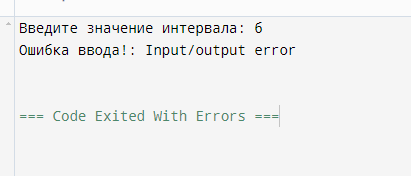


Рисунок 8 – Вывод, если введенный интервал – буква

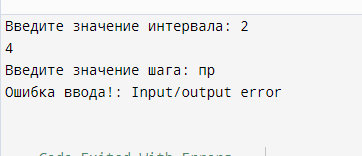


Рисунок 9 – Вывод, если введенный шаг - буква

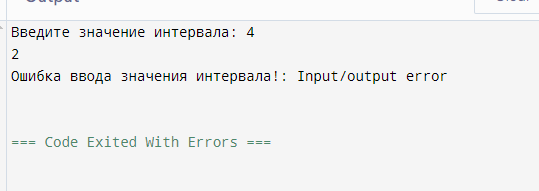


Рисунок 10 – Вывод, когда начало интервала больше конца

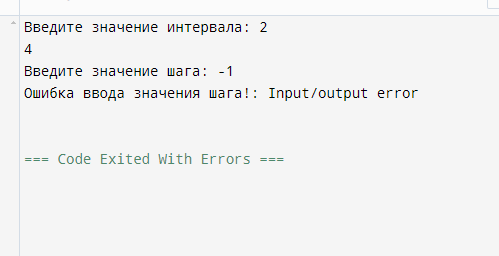


Рисунок 11 – Вывод, когда шаг – не положительное число

1. Выполнение тестовых примеров

В программе MS Excel выполнены тестовые примеры. Результаты их выполнения представлены ниже (Рисунок 12).

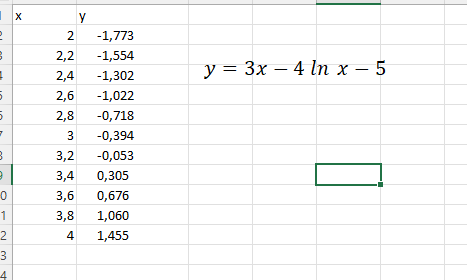


Рисунок 12 – Результат расчета функции get\_function(x)

1. Отметка о выполнении задания в веб-хостинге системы контроля версий

