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 14

Выполнил: ст. гр. ТКИ-142

Кузнецов Денис Константинович

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  – Исходные данные

|  |  |  |
| --- | --- | --- |
| Вариант | Функция | Константы |
| 14 |  |  |

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  ­ Блок-схема основного алгоритма



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



Рисунок 3 – Блок-схема функции input()



Рисунок 4 – Блок-схема функции checkStep(step)



Рисунок 5 – Блок-схема функции xCheck(x)



Рисунок 6 – Блок-схема функции getY(x)

1. Текст программы на языке C

#include <stdio.h>

#include <stdlib.h>

#include <errno.h>

#include <stdbool.h>

#include <math.h>

#include <float.h>

/\*\*

\* @brief проверка введенного значения

\* @return возвращает значение

\*/

double input(void);

/\*\*

\* @brief проверка значений интервала

\* @param iStart начало интервала

\* @param iEnd конец интервала

\* @return ошибка, если интервал некорректный

\*/

void intervalCheck(const double iStart, const double iEnd);

/\*\*

\* @brief проверяет значение шага

\* @param step значение шага

\* @return ошибка, если шаг некорректный

\*/

void stepCheck(const double step);

/\*\*

\* @brief ввод значения шага с проверкой на положительность

\* @return возвращает значение шага

\*/

double inputStep(void);

/\*\*

\* @brief проверка значения x

\* @param x значение переменной x

\* @return успех - True, неудача - False

\*/

bool xCheck(const double x);

/\*\*

\* @brief Функция, заданная формулой

\* @param x значение переменной x

\* @return возвращает значение функции в случае успеха

\*/

double getY(const double x);

/\*\*

\* @brief Точка входа в программу

\* @return Возвращает 0 в случае успеха

\*/

int main(void) {

printf("Введите начало и конец интервала: \n");

const double iStart = input();

const double iEnd = input();

intervalCheck(iStart, iEnd);

printf("Введите значение шага: ");

const double step = inputStep();

for (double x = iStart; x <= iEnd + DBL\_EPSILON; x += step) {

if (!xCheck(x)) {

printf("Недопустимое значение x: %.3lf\n", x);

} else {

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

}

}

return 0;

}

double input(void) {

double value = 0.0;

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

if (result != 1) {

errno = EIO;

perror("Input error!");

exit(EXIT\_FAILURE);

}

return value;

}

void intervalCheck(const double iStart, const double iEnd) {

if (iEnd - iStart < DBL\_EPSILON) {

errno = EIO;

perror("Неверно введён интервал!!!");

exit(EXIT\_FAILURE);

}

}

void stepCheck(const double step) {

if (step <= DBL\_EPSILON) {

errno = EIO;

perror("Неверно введён шаг!!!");

exit(EXIT\_FAILURE);

}

}

double inputStep(void) {

double step = input();

stepCheck(step);

return step;

}

bool xCheck(const double x) {

if (x >= 1) {

return false;

}

if (fabs(cos(x)) < DBL\_EPSILON) { //тангенс будет неопределён, если его косинус будет равен нулю

return false;

}

return true;

}

double getY(const double x) {

return sqrt(1 - x) - tan(x);

}

1. Результаты выполнения программы

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

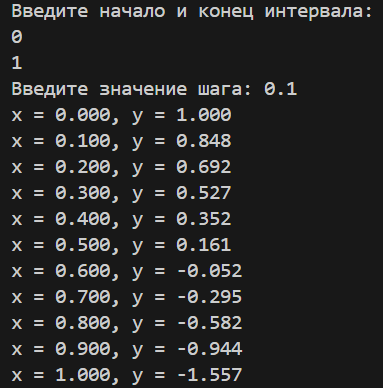


Рисунок 7 – Результаты выполнения функции с исходными данными

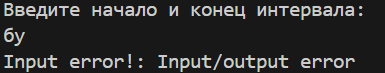


Рисунок 8 – Вывод, если пользователь ввёл букву вместо интервала

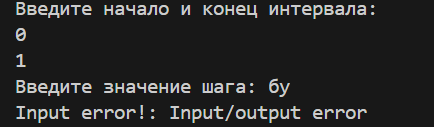


Рисунок 9 – Вывод, если пользователь ввёл букву вместо шага

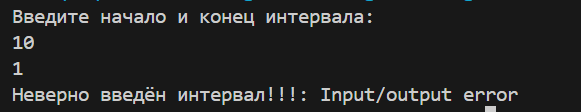


Рисунок 10 – Вывод, когда начало интервала больше конца

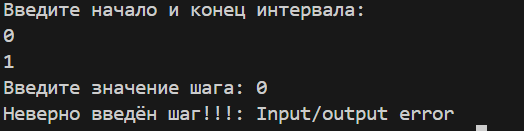


Рисунок 11 – Вывод, когда шаг – не положительное число

1. Выполнение тестовых примеров

На языке Python выполнены тестовые примеры. Результаты их выполнения представлены ниже (Рисунок 12)

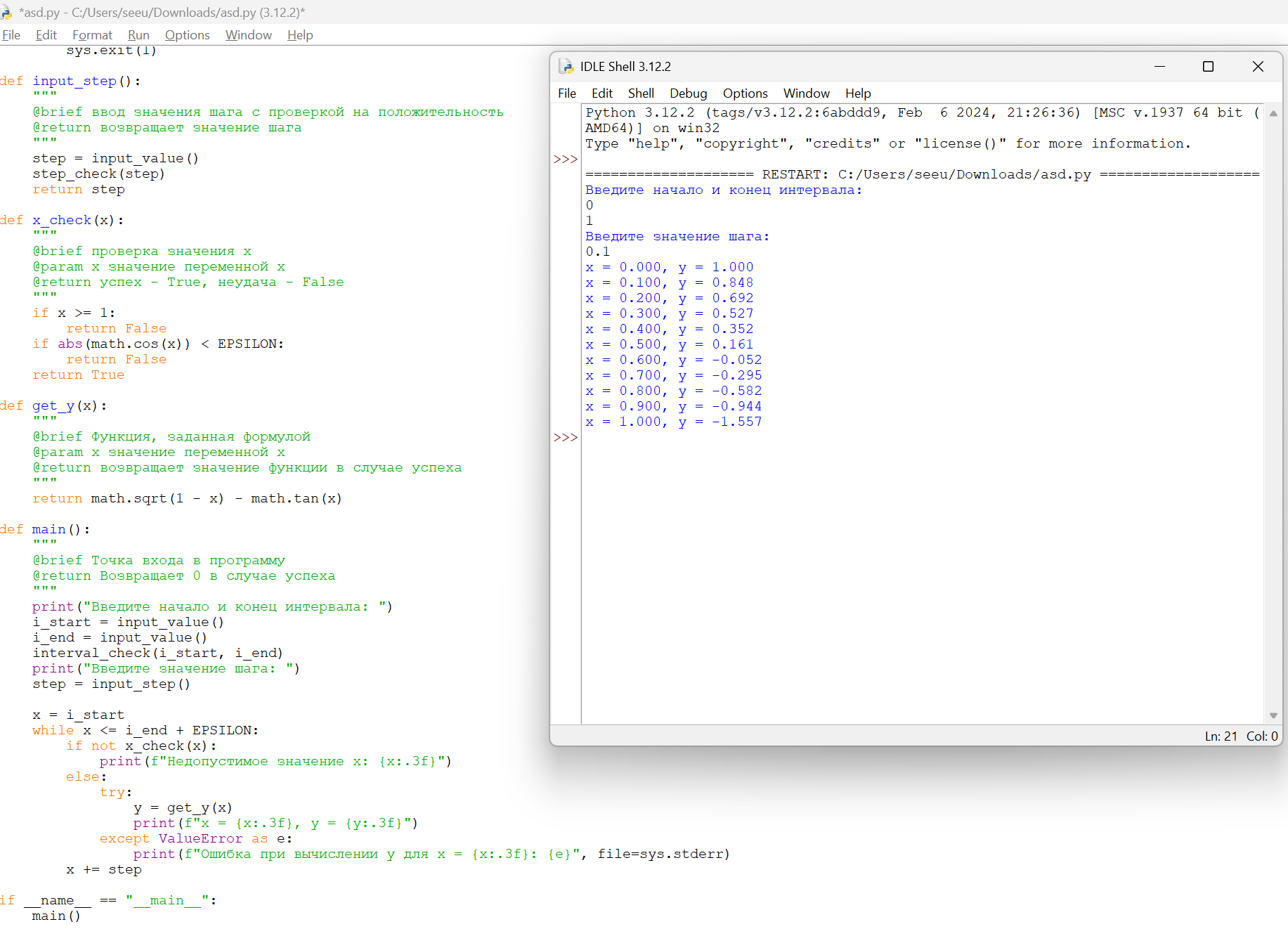


Рисунок 12 – Результат выполнения программы

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

