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 №1

По дисциплине «Языки программирования»

ВАРИАНТ 12

Выполнил: ст. гр. ТКИ-141

Комаров Констанин Сергеевич

Проверил: к.т.н., доц. Балакина Е. П.

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

Москва 2023

Оглавление

[1 Решение задачи 1–1 3](#_Toc183202115)

[1.2 Блок-схема алгоритма задание 1–1 4](#_Toc183202116)

[1.3 Текст программы на языке С задание 1–1 5](#_Toc183202117)

[1.4 Результаты выполнения программы 1–1 6](#_Toc183202118)

[1.5 Выполнение тестовых примеров задание 1–1 6](#_Toc183202119)

[1.6 Отметка о выполнении задания в веб-хостинге системы контроля версий задание 1–1 7](#_Toc183202120)

[2 Решение задачи 1–2 8](#_Toc183202121)

[2.1 Формулировка задачи 1–2 8](#_Toc183202122)

[2.2 Блок-схема алгоритма задание 1–2 8](#_Toc183202123)

[2.3 Текст программы на языке С задание 1–2 9](#_Toc183202124)

[2.4 Результаты выполнения программы 1–2 10](#_Toc183202125)

[2.5 Выполнение тестовых примеров задание 1–2 10](#_Toc183202126)

[2.6 Отметка о выполнении задания в веб-хостинге системы контроля версий задание 1–2 11](#_Toc183202127)

[3 Решение задачи 1–3 12](#_Toc183202128)

[3.1 Формулировка задачи 1–3 12](#_Toc183202129)

[3.2 Блок-схема алгоритма задание 1–3 12](#_Toc183202130)

[3.3 Текст программы на языке С задание 1–3 13](#_Toc183202131)

[3.4 Результаты выполнения программы 1–3 14](#_Toc183202132)

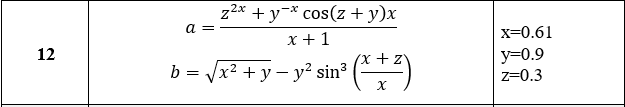
[3.5 Выполнение тестовых примеров задание 1–3 14](#_Toc183202133)

[3.6 Отметка о выполнении задания в веб-хостинге системы контроля версий задание 1–3 15](#_Toc183202134)

1. Решение задачи 1–1
   1. **Формулировка задачи 1–1**

Создать консольное приложение, вычисляющее значения переменных по представленным в таблице формулам (Таблица 1). Расчёт примера осуществить по заданным константам. Вывести на экран значения исходных данных, а также результат вычислений. Дополнить свой отчёт блок-схемой алгоритма.

Таблица 1 – Исходные данные



* 1. Блок-схема алгоритма задание 1–1

Блок-схема основного алгоритма представлена ниже (Рисунок 1). Блок-схемы функций расчета значений a и b представлены ниже (Рисунок 2).

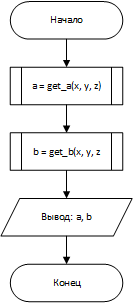


Рисунок 1 ­­–­ Блок-схема основного алгоритма

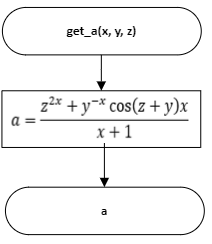
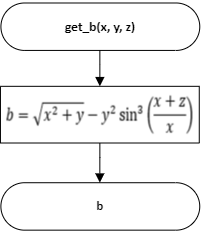
 

Рисунок 2 – Блок-схемы для вычисления a и b

* 1. Текст программы на языке С задание 1–1

#include <stdio.h>

#include <math.h>

/\*\*

\* @brief функция для вычисления значения a.

\* @param x Значение переменной x.

\* @param y Значение переменной y.

\* @param z Значение переменной z.

\* @return Результат вычисления a.

\*/

double get\_a(double x, double y, double z);

/\*\*

\* @brief функция для вычисления значения b.

\* @param x Значение переменной x.

\* @param y Значение переменной y.

\* @param z Значение переменной z.

\* @return Результат вычисления b.

\*/

double get\_b(double x, double y, double z);

/\*\*

\* @brief Точка входа в программу.

\* @return Возвращает в случае успеха.

\*/

int main() {

const double x = 0.61;

const double y = 0.9;

const double z = 0.3;

double a = get\_a(x, y, z);

double b = get\_b(x, y, z);

printf("a = %f\nb = %f", a, b);

return 0;

}

double get\_a(double x, double y, double z) {

return (pow(z, 2 \* x) + pow(y, -x) \* cos(z+y)\*x) / (x + 1);

}

double get\_b(double x, double y, double z) {

return sqrt(x \* x + y) - y \* y \* pow(sin((x + z) / x), 3);

}

* 1. Результаты выполнения программы 1–1

Результат выполнения программы в C представлен ниже (Рисунок 3).

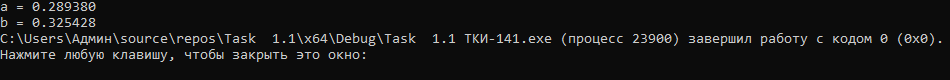


Рисунок 3 – Результат выполнения программы

* 1. Выполнение тестовых примеров задание 1–1

В программе MS Excel выполнены готовые примеры. Результаты их выполнения представлены ниже. (Рисунок 4, Рисунок 5)

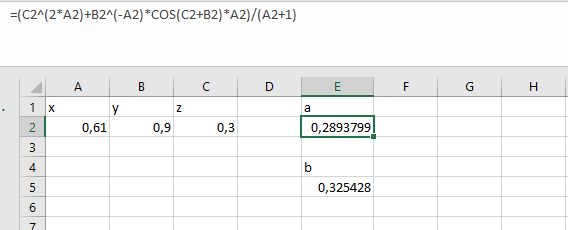


Рисунок 4 – Результат расчета переменной а

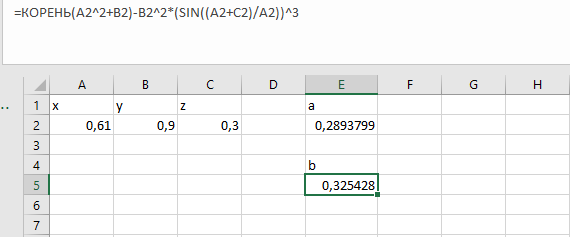


Рисунок 5 – Результат расчета переменной b

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

Ниже представлено доказательство того, что задание 1–1 было принято. (Рисунок 6)

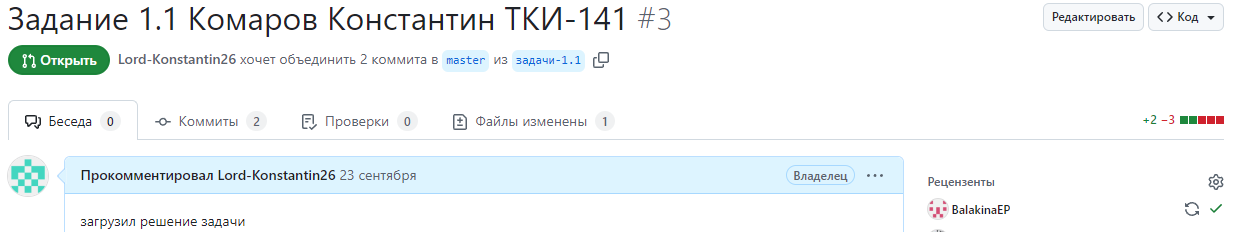


Рисунок 6 – Approve задачи 1–1

1. Решение задачи 1–2
   1. Формулировка задачи 1–2

Создать консольное приложение для решения задачи, представленной в таблице. Данные для решения вводит пользователь. Помните, что ввод необходимо проверять на правильность (только числа). Вывести результат вычислений на экран. При необходимости дополнить свой отчёт поясняющими формулами, помогающими решить задачу. Дополнить свой отчёт блок-схемой алгоритма.

Таблица 1 – Исходные данные



* 1. Блок-схема алгоритма задание 1–2

Блок-схема основного алгоритма представлена ниже (Рисунок 7). Блок-схема функции GetInch, а также проверка представлены ниже (Рисунок 8 и Рисунок 9).

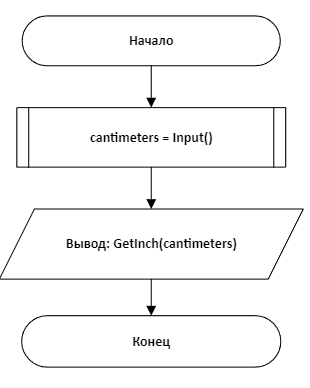


Рисунок 7 –­ Блок-схема основного алгоритма

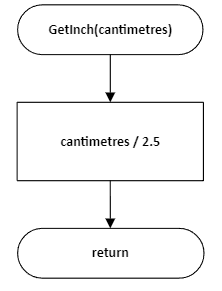


Рисунок 8 – Блок-схема функции GetInch

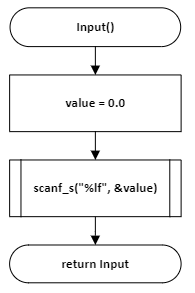


Рисунок 9 – Блок-схема проверки ввода

* 1. Текст программы на языке С задание 1–2

#include <stdio.h>

/\*

\* @brief Считывает вещественное число.

\* @return Вещественное число.

\*/

double Input(void);

/\*

\* @brief Рассчитывет количестуво дюймов.

\* @param cantimetre Количество сантиметров.

\* @return Количество дюймов

\*/

double GetInch(const double cantimeters);

/\*

\* @brief Точка входа в программу.

\* @return \c 0 в случае выполнения без ошибок.

\*/

int main()

{

puts("Введите количество сантиметров");

double cantimeters = Input();

printf\_s("Количество дюймов %lf", GetInch(cantimeters));

return 0;

}

double Input(void)

{

double value = 0.0;

scanf\_s("%lf", &value);

return value;

}

double GetInch(const double cantimeters)

{

const double toInch = 2.5;

return cantimeters / toInch;

}

* 1. Результаты выполнения программы 1–2

Результаты выполнения программы в C представлена ниже (Рисунок 10).

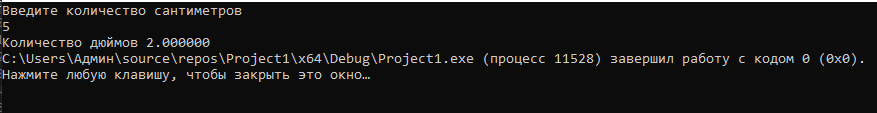


Рисунок 10 – Результаты выполнения программы

* 1. Выполнение тестовых примеров задание 1–2

В программе MS Excel выполнены тестовые примеры. Результаты их выполнения представлены ниже. (Рисунок 11)

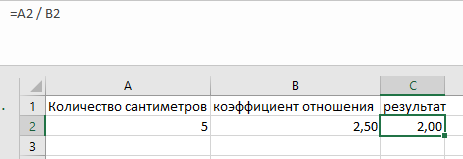


Рисунок 11 – Результат выполнения тестового примера

* 1. Отметка о выполнении задания в веб-хостинге системы контроля версий задание 1–2

Ниже представлено доказательство того, что задание 1–2 было принято. (Рисунок 12

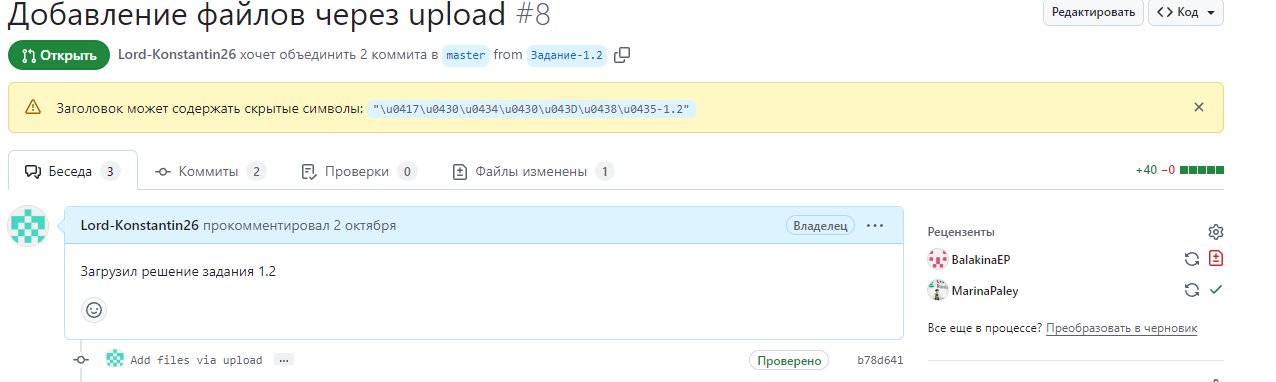
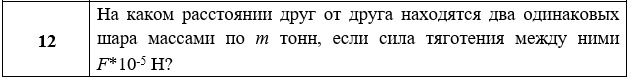


Рисунок 12 – Approve задачи 1–2

1. Решение задачи 1–3
   1. Формулировка задачи 1–3

Создать консольное приложение для решения задачи, представленной в таблице. Данные для решения вводит пользователь. Помните, что ввод необходимо проверять на правильность (только числа). Вывести результат вычислений на экран. При необходимости дополнить свой отчёт поясняющими формулами, помогающими решить задачу. Дополнить свой отчёт блок-схемой алгоритма.

Таблица 1 – Исходные данные



* 1. Блок-схема алгоритма задание 1–3

Блок-схема основного алгоритма представлена ниже (Рисунок 13). Блок-схемы функций функции GetDistance (Рисунок 14) и функции проверки ввода (Рисунок 15).

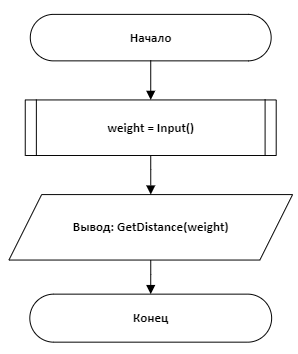


Рисунок 13 ­ Блок-схема основного алгоритма

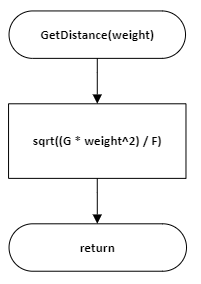


Рисунок 14 – Блок-схема функции GetDistance

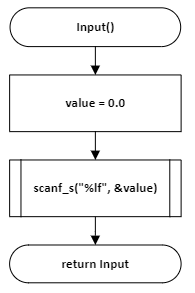


Рисунок 15 – Блок-схема функции проверки ввода

* 1. Текст программы на языке С задание 1–3

#include <stdio.h>

#include <math.h>

/\*

\* @brief Считывает вещественное число.

\* @return Вещественое число.

\*/

double Input(void);

/\*

\* @brief Рассчитывает расстояние между двумя одинаковыми шарами.

\* @param weight Масса шаров.

\* @return Расстояние между шарами.

\*/

double GetDistance(const double weight);

/\*

\* @brief Точкак входа в программу.

\* @return \c 0 в случае выполнения без ошибок.

\*/

int main()

{

puts("Введите массу шаров");

double weight = Input();

printf\_s("Расстояние %lf", GetDistance(weight));

return 0;

}

double Input(void)

{

double value = 0.0;

scanf\_s("%lf", &value);

return value;

}

double GetDistance(const double weight)

{

const double F = pow(10, -5);

const double G = 6.67 \* pow(10, -11);

return sqrt((G \* weight \* weight) / F);

}

* 1. Результаты выполнения программы 1–3

Результаты выполнения программы в C представлена ниже (Рисунок 16).

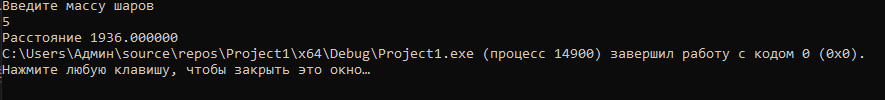


Рисунок 16 – Результат выполнения программы

* 1. Выполнение тестовых примеров задание 1–3

В программе MS Excel выполнен тестовый пример. Результат его выполнения представлен ниже. (Рисунок 17)

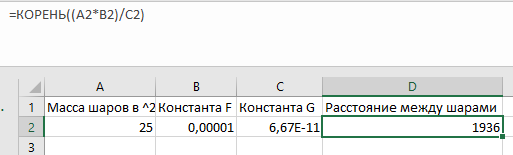


Рисунок 17 – Результат выполнения тестового примера

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

Ниже представлено доказательство того, что задание 1–3 было принято. (Рисунок 18)

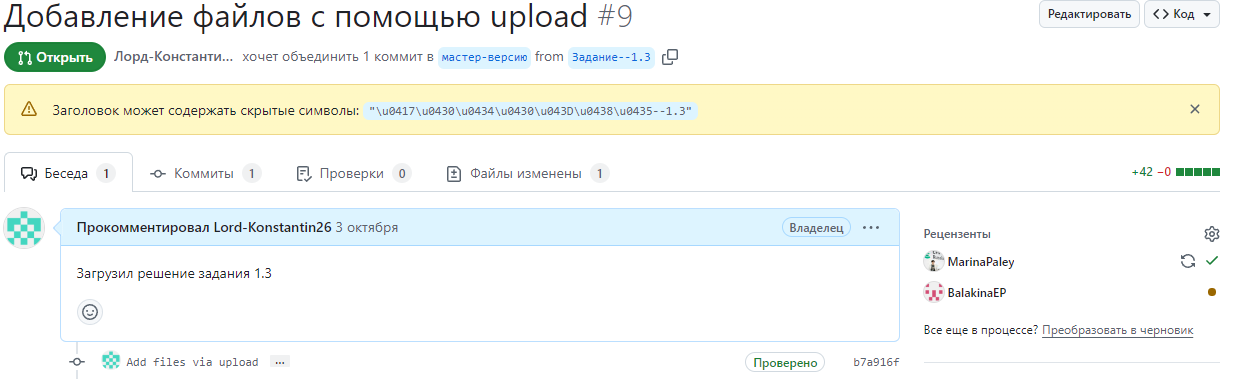


Рисунок 18 – Approve задачи 1–3