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-142

Наседкин Тимофей Филиппович

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

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

Москва 2023

**ОГЛАВЛЕНИЕ**

**1.Решение задачи 1-1\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_стр.3**

**1.1. Формулировка задачи\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_стр.3**

**1.2. Блок-схема алгоритма\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_стр.4-5**

**1.3. Код программы на языке C\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_стр.6**

**1.4. Результаты выполнения программы\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_стр.7**

**1.5. Выполнение тестовых примеров\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_стр.8**

**1.6. Отметка о выполнении задания в веб-хостинге системы контроля версий\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_стр.8**

**2.Решение задачи 1-2\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_стр.9**

**2.1. Формулировка задачи\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_стр.9**

**2.2. Блок-схема алгоритма\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_стр.10-11**

**2.3. Код программы на языке C\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_стр.12**

**2.4. Результаты выполнения программы\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_стр.13**

**2.5. Выполнение тестовых примеров \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_стр.14**

**2.6. Отметка о выполнении задания в веб-хостинге системы контроля версий\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_стр.14**

**3.Решение задачи 1-3\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_стр.15**

**3.1. Формулировка задачи\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_стр.15**

**3.2. Блок-схема алгоритма\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_стр.16-17**

**3.3. Код программы на языке C\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_стр.18**

**3.4. Результаты выполнения программы\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_стр.19**

**3.5. Выполнение тестовых примеров\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_стр.20**

**3.6. Отметка о выполнении задания в веб-хостинге системы контроля версий\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_стр.20**

* 1. **Решение задачи 1-1**
  2. **Формулировка задачи**

Создать консольное приложение, вычисляющее значения переменных по представленным в таблице формулам. Расчёт примера осуществить по заданным константам. Вывести на экран значения исходных данных, а также результат вычислений.

Таблица 1 – Исходные данные

|  |  |  |
| --- | --- | --- |
| Вариант | Формулы | Константы |
| 12 |  | x=0.61  y=0.9  z=0.3 |

* 1. **Блок-схема алгоритма**

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

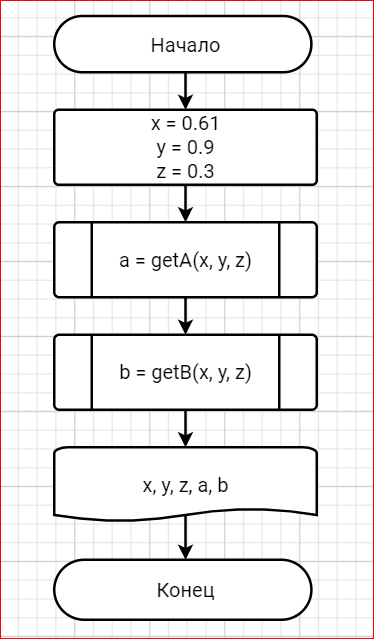


Рисунок 1 - Блок-схема основного алгоритма

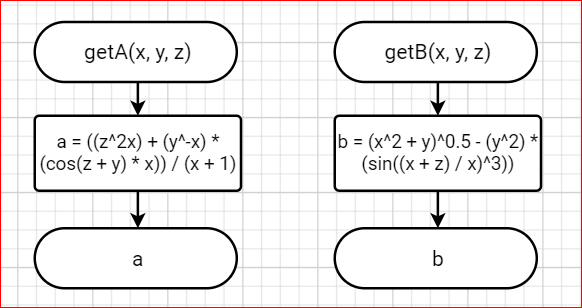


Рисунок 2 – Блок-схема используемых функций

* 1. **Код программы на языке C**

#include <math.h>

#include <stdio.h>

/\*\*

@brief Функция расчета формулы

@param x, y, z значения параметров

@return Значение вычислений

\*/

float getA(float x, float y, float z);

/\*\*

@brief Функция расчета формулы

@param x, y, z значения параметров

@return Значение вычислений

\*/

float getB(float x, float y, float z);

/\*\*

@brief Вводимое значение в программу

@return Вывод 0, если программа неверна. Иначе вывод 1

\*/

int main ()

{

float a, b;

const float x = 0.61, y = 0.9, z = 0.3;

a = getA(x, y, z);

b = getB(x, y, z);

printf ("a=%f\n b=%f\n", getA(x, y, z), getB(x, y, z));

return 0;

}

float getA(float x, float y, float z)

{

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

}

float getB(float x, float y, float z)

{

return(pow(pow (x, 2) + y, 0.5) - pow(y, 2) \* pow(sin((x + z) / x),3));

}

* 1. **Результаты выполнения программы**

Результаты выполнения программы представлены ниже (Рисунок 3).

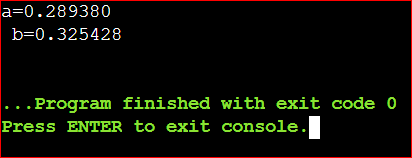


Рисунок 3 – Результаты выполнения программы

* 1. **Выполнение тестовых примеров**

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

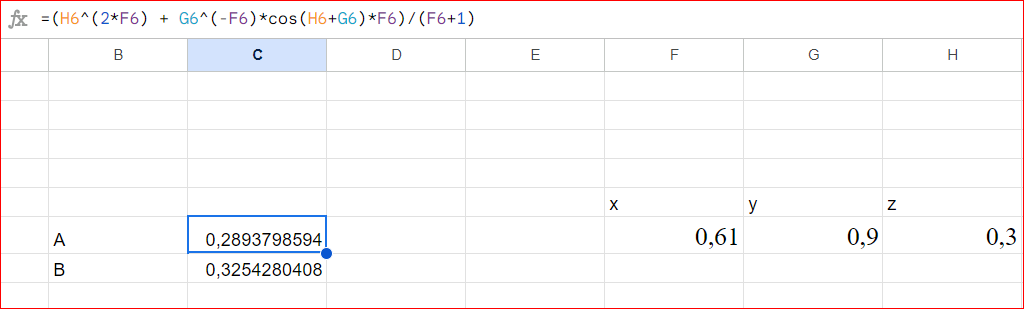


Рисунок 4 - Результат вычисления переменной a.

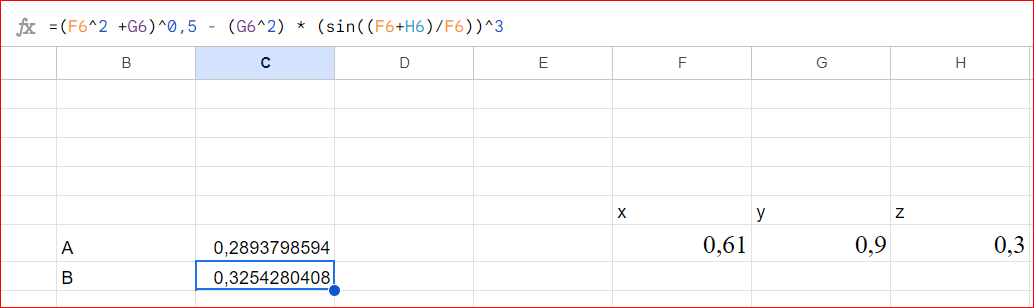


Рисунок 5 - Результат вычисления переменной b.

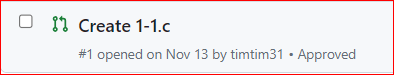
* 1. **Отметка о выполнении задания в веб-хостинге системы контроля версий** 

Рисунок 6 – Approve задачи 1.1

1. **Решение задачи 1-2**
   1. **Формулировка задачи**

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 2 – Условие задачи 1.2

|  |  |
| --- | --- |
| Вариант | Задачи |
| 12 | Длина выражена в сантиметрах. Выразить ее в дюймах.  (1 дюйм=2.5 см) |

1. **Блок-схема алгоритма**

Блок-схема основного алгоритма представлена ниже (Рисунок 1). Блок-схемы дополнительных функций представлены ниже (Рисунок 2, Рисунок 3).

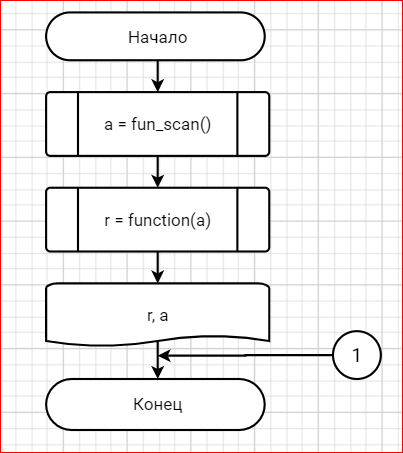


Рисунок 1 - Блок-схема основного алгоритма

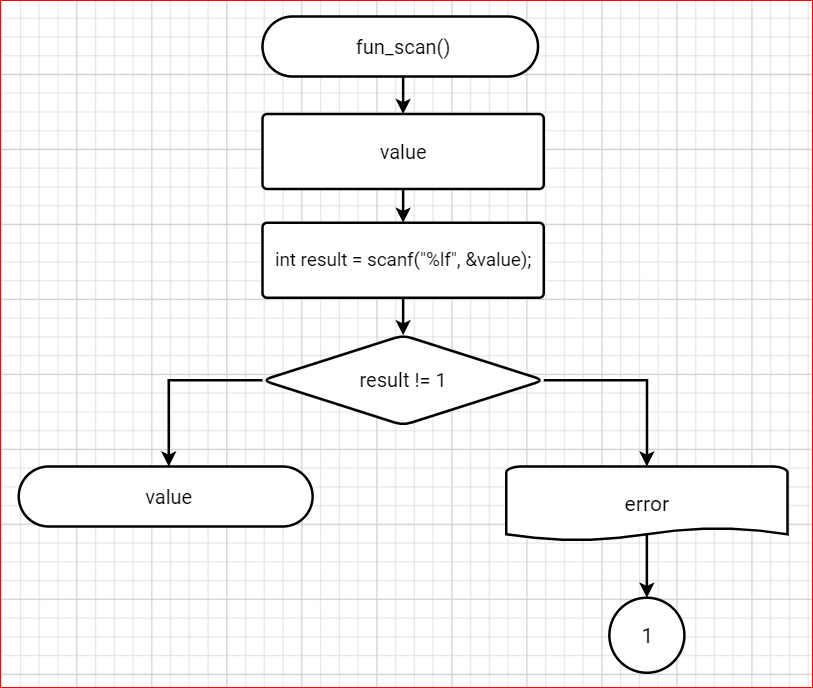


Рисунок 2 - Блок-схема функции fun\_scan()

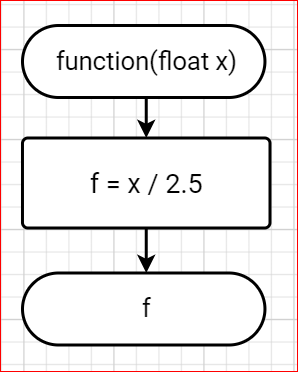


Рисунок 3- Блок-схема функции function()

1. **Код программы на языке C**

#include <stdio.h>

#include <stdlib.h>

/\*\*

@brief Функция расчета формулы

@param x значение параметра

@return Значение вычислений

\*/

float function(float x);

/\*\*

@brief Функция проверки ввода на число

@return Результат проверки

\*/

double fun\_scan();

/\*\*

@brief Вводимое значение для входа в программу

@return Вывод 0, если программа неверна. Иначе вывод 1

\*/

int main()

{

float a = fun\_scan();

printf("r=%f\n", function(a));

return 0;

}

float function(float x)

{

return x / 2.5;

}

double fun\_scan()

{

double value;

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

if (result != 1)

{

printf("error io");

abort();

}

return value;

}

* 1. **Результаты выполнения программы**

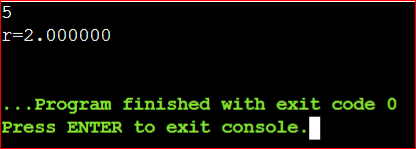
Результаты выполнения программы представлены ниже (Рисунок 3)

Рисунок 3 - результаты выполнения программы

* 1. **Выполнение тестовых примеров**

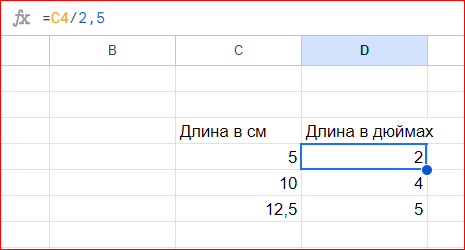
В программе MS Excel выполнены тестовые примеры. Результаты их выполнения представлены ниже (Рисунок 4)

Рисунок 4 - Результат вычислений программы

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

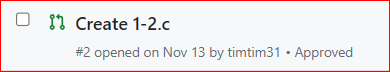


Рисунок 5 – Approve задачи 1.2

1. **Решение задачи 1-3**
2. **Формулировка задачи**

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 3 – Условие задачи 1.3

|  |  |
| --- | --- |
| **Вариант** | **Задача** |
| **12** | На каком расстоянии друг от друга находятся два одинаковых шара массами по *m* тонн, если сила тяготения между ними  *F*\*10-5 Н? |

1. **Блок-схема алгоритма**

Блок-схема основного алгоритма представлена ниже (Рисунок 1). Блок-схемы дополнительных функций представлены ниже (Рисунок 2, Рисунок 3).

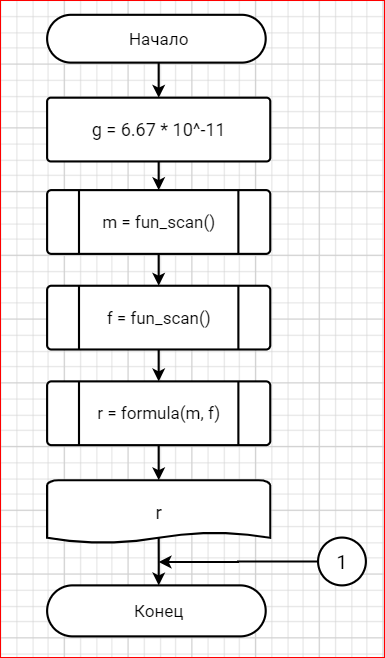


Рисунок 1 - Блок-схема основного алгоритма

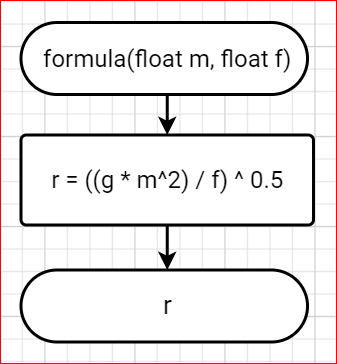


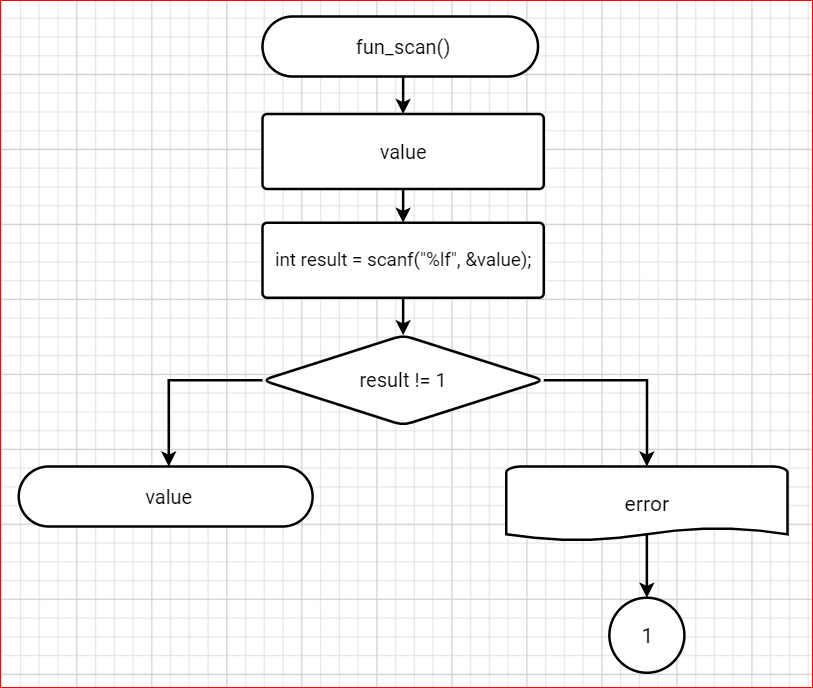
Рисунок 2- Блок-схема функции formula()

Рисунок 3 - Блок-схема функции fun\_scan()

1. **Код программы на языке C**

#include <stdio.h>

#include <math.h>

#include <stdlib.h>

#include <float.h>

/\*\*

@brief Функция расчета физической формулы

@param m, f значения вводимых параметров

@return Значение вычисления формулы

\*/

float formula(float m, float f);

/\*\*

@brief Функция проверки ввода на число

@return Результат проверки

\*/

double fun\_scan();

/\*\*

@brief Вводимое значение для входа в программу

@return Вывод 0, если программа неверна. Иначе вывод 1

\*/

int main()

{

float m = fun\_scan();

float f = fun\_scan();

printf("r=%f\n", formula(m, f));

return 0;

}

float formula(float m, float f)

{

const float g = 6.67 \* pow(10, -11);

return (pow(((g \* m \* m) / f), 0.5));

}

double fun\_scan()

{

double value;

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

if ((result != 1) || (result < DBL\_EPSILON))

{

puts("error io");

abort();

}

return value;

}

1. **Результаты выполнения программы**

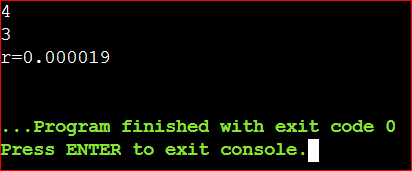
Результаты выполнения программы представлены ниже (Рисунок 3)

Рисунок 3 - результаты выполнения программы

1. **Выполнение тестовых примеров**

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

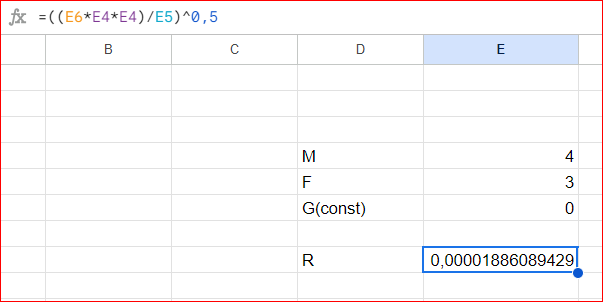


Рисунок 4 - Результат вычислений программы

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

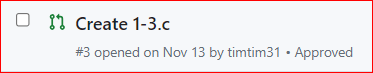


Рисунок 5 – Approve задачи 1.3