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 9

Выполнил: ст. гр. ТКИ - 141

Сивунов Пётр Сергеевич

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

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

Москва 2024

* + 1. Формулировка задания

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. Дополнить свой отчёт блок-схемой алгоритма.

Таблица  – Исходные данные

|  |  |  |
| --- | --- | --- |
| Вариант | Формулы | Константы |
| 8 |  | x=0.2  y=0.004  z=1.1 |

* + 1. Блок-схема алгоритма

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

Изображение выглядит как текст, диаграмма, снимок экрана, линия

Автоматически созданное описание

Рисунок  ­ Блок-схема основного алгоритма

Изображение выглядит как текст, диаграмма, круг, линия

Автоматически созданное описание

Рисунок  – Блок-схема используемых функций

* + 1. Текст программы на языке C

#include <math.h>

#include <stdio.h>

/\*\*

\* @brief Расчитывает значение A

\* @param x const double

\* @param y const double

\* @param z const double

\* @return Вычисляет значение A

\*/

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

/\*\*

\* @brief Расчитывает значение B

\* @param x const double

\* @param y const double

\* @param z const double

\* @return Вычисляет значение B

\*/

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

/\*\*

\* @brief Точка входа в программу

\* @return 0

\*/

int main()

{

const double x = 0.2;

const double y = 0.004;

const double z = 1.1;

printf("Task1 %f \n", get\_a(x, y, z));

printf("Task2 %f \n", get\_b(x, y, z));

return 0;

}

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

{

return pow(sin(pow(x, 2) + pow(z, 2)), 3) - sqrt(x / y);

}

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

{

return pow(x, 2) / z + pow(cos(x + y), 3);

}

* + 1. Результаты выполнения программы

Результаты выполнения программы представлены ниже (Рисунок 3).

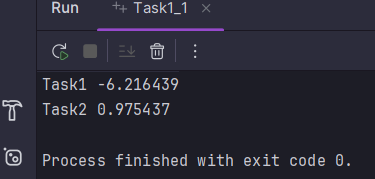


Рисунок  – Результаты выполнения программы

* + 1. Выполнение тестовых примеров

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

Изображение выглядит как текст, снимок экрана, число, Шрифт

Автоматически созданное описание

Рисунок  – Результат расчета переменной a

Изображение выглядит как текст, снимок экрана, число, Шрифт

Автоматически созданное описание

Рисунок  – Результат расчета переменной bs

* + 1. Формулировка задания

Создать консольное приложение для решения задачи, представленной в таблице. Данные для решения вводит пользователь. Вывести результат вычислений на экран. При необходимости дополнить свой отчёт поясняющими формулами, помогающими решить задачу. Дополнить свой отчёт блок-схемой алгоритма.

Таблица  – Исходные данные

|  |  |
| --- | --- |
| Вариант | Задачи |
| 9 | Известен объем информации в байтах. Выразить его в мегабайтах и гигабайтах. |

* + 1. БЛОК-СХЕМА АЛГОРИТМА

Изображение выглядит как текст, диаграмма, снимок экрана, линия

Автоматически созданное описание

Рисунок 6 – Блок схема функции main

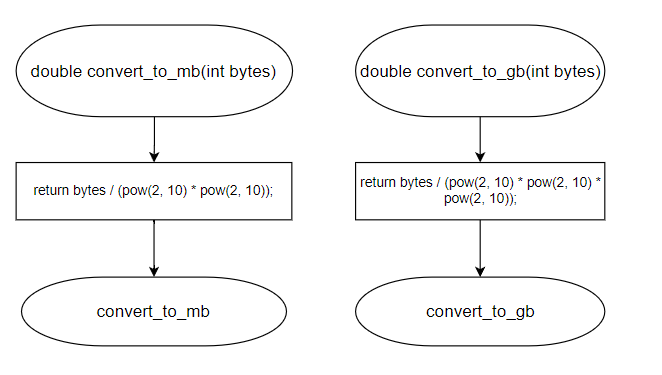


Рисунок 7- Блок схема функций перевода в МБ и ГБ

Изображение выглядит как текст, диаграмма, круг, линия

Автоматически созданное описание

Рисунок 8 – Функция ввода

* + 1. ТЕКСТ ПРОГРАММЫ НА ЯЗЫКЕ C

#include <locale.h>  
#include <math.h>  
#include <stdio.h>  
#include <stdlib.h>  
  
*/\*\*  
 \* @brief Переводит байты в мегабайты  
 \* @param bytes - количество байт  
 \* @return double - количество мегабайт  
 \*/*double convert\_to\_mb(int bytes);  
  
*/\*\*  
 \* @brief Переводит байты в гигабайты  
 \* @param bytes - количество байт  
 \* @return double - количество гигабайт  
 \*/*double convert\_to\_gb(int bytes);  
  
*/\*\*  
 \* @brief Функция ввоода целого числа  
 \* @return Целое число  
 \*/*int input();  
  
*/\*\*  
 \* @brief Точка входа в программу  
 \* @return 0  
 \*/*int main(int argc, char\* argv[])  
{  
 setlocale(LC\_ALL, "Rus");  
 puts("Введите байты: ");  
 int enter\_bytes = input();  
  
 printf("МБ: %lf\n", convert\_to\_mb(enter\_bytes));  
 printf("ГБ: %lf\n", convert\_to\_gb(enter\_bytes));  
  
 return 0;  
}  
  
  
int input()  
{  
 int value = 0;  
 if (scanf\_s("%d", &value) != 1)  
 {  
 abort();  
 }  
 return value;  
}  
  
double convert\_to\_mb(int bytes)

{

return bytes / (pow(2, 10) \* pow(2, 10));

}

double convert\_to\_gb(int bytes)

{

return bytes / (pow(2, 10) \* pow(2, 10) \* pow(2, 10));

}

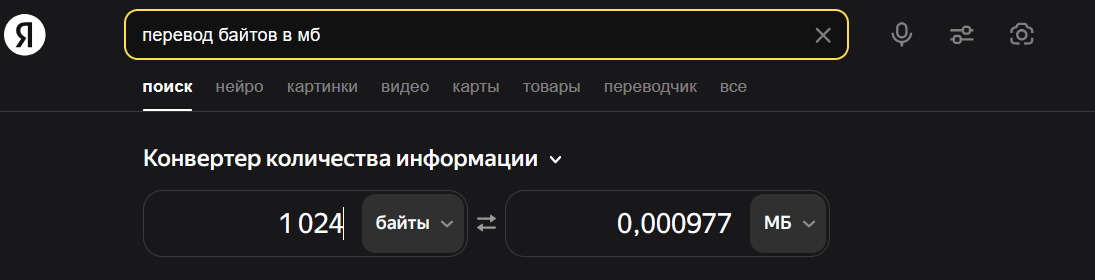
* + 1. РЕЗУЛЬТАТЫ ВЫПОЛНЕНИЯ ПРОГРАММЫ

Результаты выполнения программы представлены ниже (Рисунок 3).

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

Рисунок 9 – Результаты выполнения программы

* + 1. ВЫПОЛНЕНИЕ ТЕСТОВЫХ ПРИМЕРОВ

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описаниеРисунок 10 – Тест перевода байтов в МБ

Рисунок 11 – Тест перевода байтов в ГБ

* + 1. ФОРМУЛИРОВКА ЗАДАНИЯ

Создать консольное приложение для решения задачи, представленной в таблице. Данные для решения вводит пользователь. Вывести результат вычислений на экран. При необходимости дополнить свой отчёт поясняющими формулами, помогающими решить задачу. Дополнить свой отчёт блок-схемой алгоритма.

Таблица 3 – Исходные данные

|  |  |
| --- | --- |
| Вариант | Задачи |
| 9 | Три резистора (*R1 R2 R3*) соединены последовательно. Найдите сопротивление соединения. |

* + 1. БЛОК-СХЕМА АЛГОРИТМА

Изображение выглядит как текст, диаграмма, чек, линия

Автоматически созданное описание

Рисунок 12 – Блок схема main

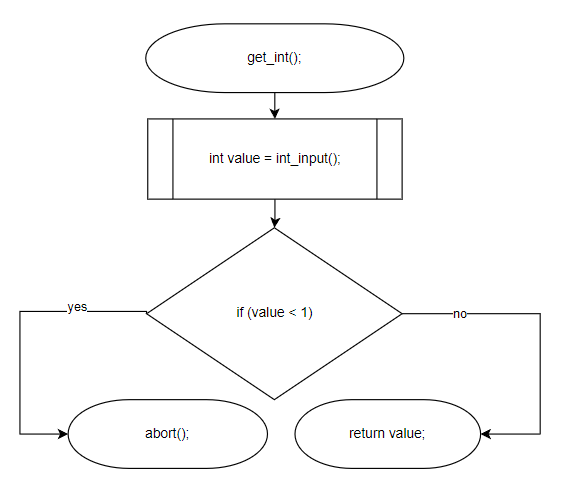


Рисунок 13 – Блок схема get\_int

Изображение выглядит как диаграмма, текст, линия, круг

Автоматически созданное описание

Рисунок 14 – Блок схема int\_input

Изображение выглядит как текст, диаграмма, снимок экрана, линия

Автоматически созданное описание

Рисунок 15 – Блое схема try\_allocate\_memory

Изображение выглядит как текст, диаграмма, снимок экрана, Шрифт

Автоматически созданное описание

Рисунок 16 – Блок схема count\_r

* + 1. ТЕКСТ ПРОГРАММЫ НА ЯЗЫКЕ C

#include <stdio.h>  
#include <stdlib.h>  
  
*/\*\*  
 \* @brief Вычисляет общее сопротивление последовательно соединенных резисторов  
 \* @param resistors массив float   
 \* @param size Количество сопротивлений  
 \* @return   
 \*/*float count\_r(const float\* resistors, int size);  
  
*/\*\*  
 \* @ Функция ввода целого числа  
 \* @return int  
 \*/*int int\_input();  
  
*/\*\*  
 \* @brief Функция ввода вещественного числа  
 \* @return float  
 \*/*float float\_input();  
  
*/\*\*  
 \* @brief Функция проверки целого числа  
 \* @return int  
 \*/*int get\_int();  
  
*/\*\*  
 \* @brief Пытается выделить память  
 \* @param size размер массива  
 \* @return   
 \*/*float\* try\_allocate\_memory(int size);  
  
  
*/\*\*  
 \* @brief Точка входа в программу  
 \* @return 0  
 \*/*int main(int argc, char\* argv[])  
{  
  
 puts("Enter the number of resistors: ");  
 int size = get\_int();  
  
 float\* resistors = try\_allocate\_memory(size);  
   
 for (int i = 0; i < size; i++)  
 {  
 printf("Enter resistor #%d: ", i + 1);  
 resistors[i] = float\_input();  
 }  
   
 float resistance = count\_r(resistors, size);  
 printf("Total resistance: %f\n", resistance);  
   
 free(resistors);  
   
 return 0;  
}  
  
  
int get\_int()  
{  
 int value = int\_input();  
 if (value < 1)  
 {  
 abort();  
 }  
 return value;  
}  
  
int int\_input()  
{  
 int value = 0;  
 if (scanf\_s("%d", &value) != 1)  
 {  
 abort();  
 }  
 return value;  
}  
  
float float\_input()  
{  
 float value = 0;  
 if (scanf\_s("%f", &value) != 1)  
 {  
 abort();  
 }  
 return value;  
}  
  
float\* try\_allocate\_memory(int size)  
{  
 float\* array = malloc(size \* sizeof(float));  
 if (array == NULL)  
 {  
 abort();  
 }  
 return array;  
}  
  
float count\_r(const float\* resistors, int size)  
{  
 float sum = 0;  
 for (int i = 0; i < size; i++)  
 {  
 sum += resistors[i];  
 }  
 return sum;  
}

* + 1. РЕЗУЛЬТАТЫ ВЫПОЛНЕНИЯ ПРОГРАММЫ

Результаты выполнения программы представлены ниже (Рисунок 3).

Изображение выглядит как текст, снимок экрана, Шрифт, алгебра

Автоматически созданное описание-

Рисунок  – Результаты выполнения программы

**Approve скриншоты**

Изображение выглядит как программное обеспечение, текст, Мультимедийное программное обеспечение, снимок экрана

Автоматически созданное описание

Изображение выглядит как программное обеспечение, Мультимедийное программное обеспечение, текст, снимок экрана

Автоматически созданное описание