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 №2

По дисциплине «Языки программирования»

ВАРИАНТ 5

Выполнил: ст. гр. ТКИ-142

Скрипников Егор 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 для решения задачи, представленной в таблице (Таблица 1), с использованием перечислимого типа. Выбор формулы вычисления зависит от пользователя. Данные для решения задачи так же вводит пользователь. Ввод необходимо проверять на правильность. Все результаты вывести на экран. Дополнить свой отчёт блок-схемой алгоритма.

Таблица  – Исходные данные

|  |  |
| --- | --- |
| Вариант | Задачи |
| 5 | Заданы стороны прямоугольника. Определить его:   * периметр; * площадь; * длину диагонали. |

1. Блок-схема алгоритма

Блок-схема основного алгоритма представлена ниже (Рисунок 1). Блок-схемы функций расчета значений периметра, площади, диагонали и ввода представлены ниже (Рисунок 2, 3, 4, 5).



Рисунок  ­ Блок-схема основного алгоритма



Рисунок  – Блок-схема функции printMenu()



Рисунок 3 – Блок-схема функции inputDouble()



Рисунок 4 – Блок-схема функции inputInt()



Рисунок 5 – Блок-схема функций getPerimeter(x,y), getArea(x,y), getDiagonal(x,y)

1. Текст программы на языке C

#include <stdio.h>

#include <stdlib.h>

#include <math.h>

#include <errno.h>

*/\*\**

*\* @brief Вызывает меню*

*\* @return Возвращает значение выбранный параметр*

*\*/*

void printMenu(void);

*/\*\**

*\* @brief Ввод полученного вещественного значения*

*\* @return Возвращает проверенное вещественное число*

*\*/*

double inputDouble(void);

*/\*\**

*\* @brief Ввод полученного целого значения*

*\* @return Возвращает полученное целое значение*

*\*/*

int inputInt(void);

*/\*\**

*\* @brief Рассчитывает значение периметра по данным сторонам*

*\* @param x значение ширины*

*\* @param y значение длины*

*\* @return Возвращает значение периметра*

*\*/*

double getPerimeter(double const x, double const y);

*/\*\**

*\* @brief Рассчитывает значение площади по данным сторонам*

*\* @param x значение ширины*

*\* @param y значение длины*

*\* @return Возвращает значение площади*

*\*/*

double getArea(double const x, double const y);

*/\*\**

*\* @brief Рассчитвает значение диагонали по данным сторонам*

*\* @param x значение ширины*

*\* @param y значение длины*

*\* @return Возвращает значение диагонали*

*\*/*

double getDiagonal(double const x, double const y);

*/\*\**

*\* @param perimeter периметр прямоугольника*

*\* @param area площадь прямоугольника*

*\* @param diagonal диагональ прямоугольника*

*\*/*

enum request{perimeter, area, diagonal};

*/\*\**

*\* @brief Точка входа в программу*

*\* @return Возвращает 0 в случае успеха*

*\*/*

int main(void)

{

    double const x = inputDouble();

    double const y = inputDouble();

    printMenu();

    int query = inputInt();

    switch (query)

    {

    case perimeter:

        printf("Perimeter of rectangle = %.2lf\n", getPerimeter(x, y));

        break;

    case area:

        printf("Area of the rectangle = %.2lf\n", getArea(x, y));

        break;

    case diagonal:

        printf("Diagonal of the rectangle = %.2lf\n", getDiagonal(x, y));

        break;

    default:

        errno = ERANGE;

        perror("Function does not exist\n");

        exit(EXIT\_FAILURE);

    }

    return 0;

}

void printMenu(void)

{

    printf("Perimeter - %d\n", perimeter);

    printf("Area - %d\n", area);

    printf("Diagonal - %d\n", diagonal);

}

double inputDouble(void)

{

    double value = 0.0;

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

    if (result != 1)

    {

        errno = EIO;

        perror("Input error\n");

        exit(EXIT\_FAILURE);

    }

    return value;

}

int inputInt(void)

{

    int c = 0;

    int check = scanf\_s("%d", &c);

    if (check != 1)

    {

        errno = EIO;

        perror("Input error\n");

        exit(EXIT\_FAILURE);

    }

    return c;

}

double getPerimeter(double const x, double const y)

{

    return 2 \* (x + y);

}

double getArea(double const x, double const y)

{

    return x \* y;

}

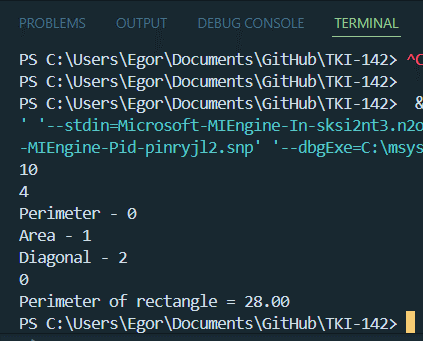
double getDiagonal(double const x, double const y)

{

    return sqrt(x \* x + y \* y);

}

1. Результаты выполнения программы

Результаты выполнения программы представлены ниже (Рисунок 6, 7, 8, 9, 10, 11, 12, 13).

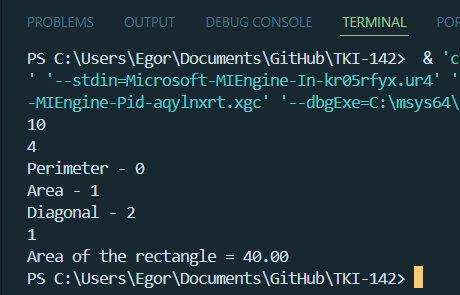
Рисунок 6 – Результаты вычисления периметра прямоугольника

Рисунок 7 – Результаты вычисления площади прямоугольника

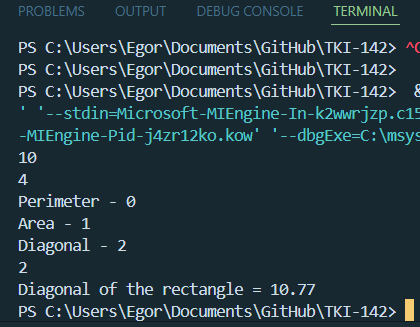


Рисунок 8 – Результаты вычисления диагонали прямоугольника

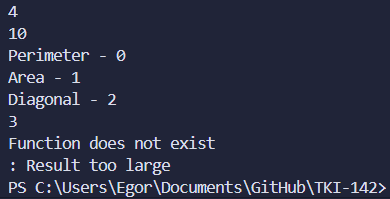


Рисунок 9 – Вывод, когда выбор не входит в указанные значения

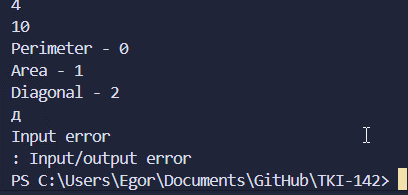


Рисунок 10 – Вывод, когда выбор – буква

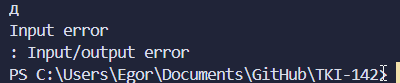


Рисунок 11 – Вывод, когда x – буква

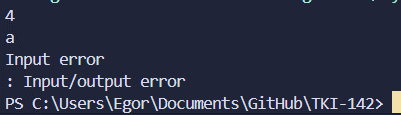


Рисунок 12 – Вывод, когда y – буква

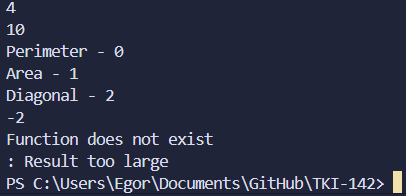
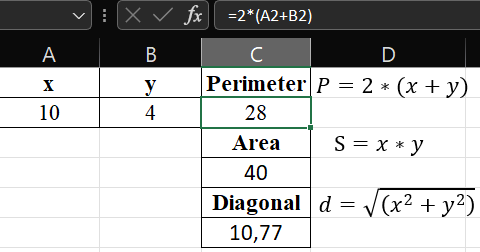


Рисунок 13 – Вывод, когда выбор – отрицательное число

1. Выполнение тестовых примеров

В программе MS Excel выполнены тестовые примеры. Результаты их выполнения представлены ниже (Рисунок 14, 15, 16).



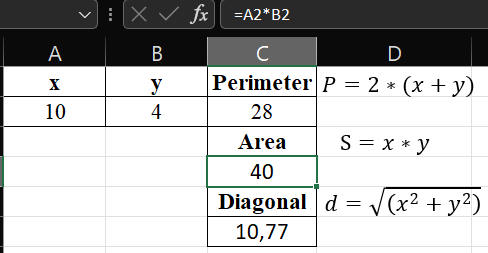
Рисунок 14 – Результат расчета функции getPerimeter

Рисунок 15 – Результат расчет функции getArea

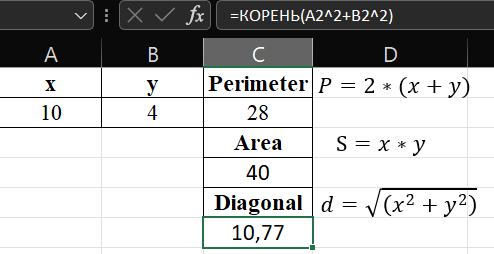


Рисунок 16 – Результат расчета функции getDiagonal

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

