МИНИСТЕРСТВО НАУКИ и высшего образования

**РОССИЙСКОЙ ФЕДЕРАЦИИ**

Федеральное государственное автономное образовательное учреждение

высшего образования

«Санкт-Петербургский политехнический университет Петра Великого»

(ФГАОУ ВО «СПбПУ»)

**Институт среднего профессионального образования**

**Отчёт по лабораторной работе № 2**

**по учебной дисциплине «Междисциплинарный курс 01.01 Разработка программных модулей»**

**Тема: "** **Определение и вызов функций. Передача параметров и возврат результатов."**

Выполнил(а) студент(ка)

специальности 09.02.07

Информационные технологи и

программирование

III курса группы 32919/3

Канищева Анастасия

Васильевна

Преподаватель

Ильин Юрий Петрович

Санкт-Петербург,

2024г.

**Лабораторная работа № 2**

**Тема:** Определение и вызов функций. Передача параметров и возврат результатов.

**Цель работы:** изучить правила определения и вызова функций в языке C#, получить практические навыки разработки спецификаций функций по обработке простых типов данных и функций по обработке структурированных типов данных, и создания программ, использующих функции программиста.

*Указания:*

Требуется разработать 1 проект, каждая программа − консольное приложение на языке C#, в среде программирования Visual Studio, использующая дополнительные функции.

**Задача 19**

Разработать подпрограмму для вычисления площади треугольника по двум сторонам и углу между ними, и подпрограмму для вычисления его высот. Угол задается в градусах.

**Математическая модель**

Входные данные:

**side1, side2** – число с плавающей точкой (double), вводится с клавиатуры, является сторонами треугольника

**angle** - число с плавающей точкой (double), вводится с клавиатуры является углом между сторонами треугольника

Выходные данные:

**S** - типа double – с плавающей точкой (площадь);

**heightl** = {heightl, i=0,1, 2} высоты к стороне модификатор out c типом double, т.е. параметр, передаваемый по ссылке

Связь:

* Площадь треугольника, при выполнении формулы

угол в радианах = angle \* PI / 180

SТреугольника = 0.5 \* side1 \* side2 \* sin(угол в радианах);

* Высота треугольника к первой стороне, при выполнении формулы

height1 = side2\* sin(угол в радианах);

* Высота ко второй стороне, при выполнении формулы

height2 = side1\* sin(угол в радианах);

* Высота к третей стороне, при выполнении формулы:

height2 = 2 \* SТреугольника / ТретьяСторона

**Блок схема:**



Рисунок 1 - Блок схема

**Код программы:**

using System;

class Program

{

// Пользовательское исключение для некорректного угла

public class InvalidAngleException : Exception

{

public InvalidAngleException(string message) : base(message) { }

}

// Подпрограмма для вычисления площади треугольника по двум сторонам и углу между ними

static double SТреугольника(double side1, double side2, double angle)

{

// Проверка корректности угла

if (angle <= 0 || angle >= 180)

{

throw new InvalidAngleException("Угол должен быть в диапазоне от 0 до 179 градусов.");

}

// Преобразование угла из градусов в радианы

double уголВРадианах = angle \* Math.PI / 180;

// Вычисление площади по формуле

return Math.Round(0.5 \* side1 \* side2 \* Math.Sin(уголВРадианах), 2); // Округление до 2 знаков после запятой

}

// Подпрограмма для вычисления третьей стороны по теореме косинусов

static double ТретьяСторона(double side1, double side2, double angle)

{

// Преобразование угла из градусов в радианы

double уголВРадианах = angle \* Math.PI / 180;

// Вычисление третьей стороны по теореме косинусов

return Math.Round(Math.Sqrt(Math.Pow(side1, 2) + Math.Pow(side2, 2) - 2 \* side1 \* side2 \* Math.Cos(уголВРадианах)), 2);

}

// Подпрограмма для вычисления высот треугольника

static void HТреугольника(double side1, double side2, double angle, out double height1, out double height2, out double height3)

{

// Проверка корректности угла

if (angle <= 0 || angle >= 180)

{

throw new InvalidAngleException("Угол должен быть в диапазоне от 0 до 179 градусов.");

}

// Преобразование угла из градусов в радианы

double уголВРадианах = angle \* Math.PI / 180;

// Вычисление высот по формулам

height1 = Math.Round(side2 \* Math.Sin(уголВРадианах), 2); // Округление до 2 знаков после запятой

height2 = Math.Round(side1 \* Math.Sin(уголВРадианах), 2); // Округление до 2 знаков после запятой

// Вычисление площади треугольника

double площадь = SТреугольника(side1, side2, angle);

// Вычисление третьей стороны

double side3 = ТретьяСторона(side1, side2, angle);

// Высота к третьей стороне: h3 = (2 \* площадь) / side3

height3 = Math.Round((2 \* площадь) / side3, 2);

}

static void Main(string[] args)

{

double side1, side2, angle, height1, height2, height3;

try

{

Console.WriteLine("Введите длину первой стороны треугольника:");

side1 = Double.Parse(Console.ReadLine());

Console.WriteLine("Введите длину второй стороны треугольника:");

side2 = Double.Parse(Console.ReadLine());

Console.WriteLine("Введите угол между сторонами (в градусах):");

angle = Double.Parse(Console.ReadLine());

// Вычисление площади треугольника

double S = SТреугольника(side1, side2, angle);

Console.WriteLine("Площадь треугольника: {0}", S);

// Вычисление высот треугольника

HТреугольника(side1, side2, angle, out height1, out height2, out height3);

Console.WriteLine("Высота к первой стороне: {0}", height1);

Console.WriteLine("Высота ко второй стороне: {0}", height2);

Console.WriteLine("Высота к третьей стороне: {0}", height3);

}

catch (InvalidAngleException ex)

{

Console.WriteLine($"Ошибка ввода угла: {ex.Message}");

}

catch (Exception ex)

{

Console.WriteLine($"Произошла ошибка: {ex.Message}");

}

}

}

**Вывод:**

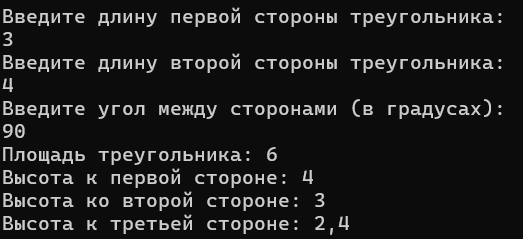


Рисунок 2 - Результат

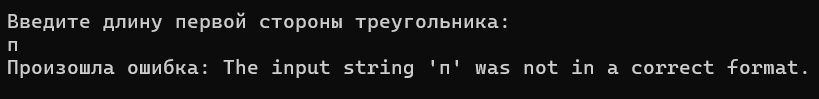


Рисунок 3 - Результат с ошибкой

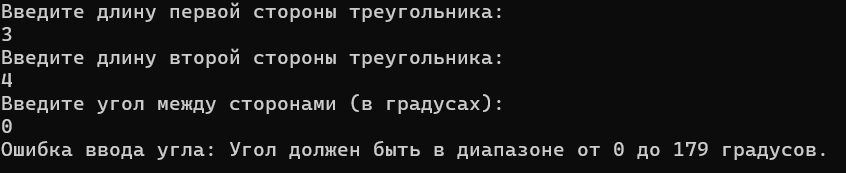


Рисунок 4 - Результат с ошибкой

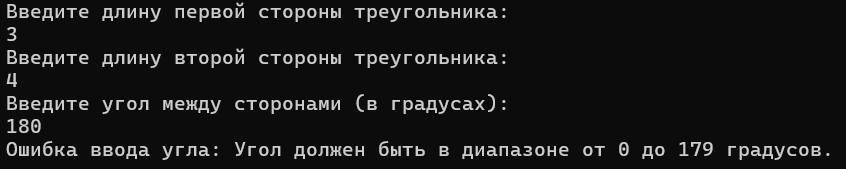


Рисунок 5 - Результат с ошибкой