МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

Учреждение образования” Гомельский государственный университет им Ф. Скорины”

Отчет по лабораторной работе

**КФ-Программирование-1-1-06 ПРОГРАММИРОВАНИЕ СОБСТВЕННЫХ МЕТОДОВ НА ЯЗЫКЕ C#**

Выполнил:

Студент группы КФ-17

Гуревич П.А.

Проверил Дей Е.А.

Гомель 2022

**Цель работы:** изучение правил и выработка навыков составления и использования собственных функций при разработке программ на языке С#

ЗАДАЧА 1. Создать новый проект и повторить все примеры, добавляя их (с учетом уже набранных команд) в одну программу.

Изменить численные константы в тексте программы и выполнить программу 3 раза с различными входными наборами данных.

Результаты включить в отчет.

ЗАДАЧА 2. Составить программу для вычисления и вывода на экран требуемого результата. (Варианты приведены в Приложении 1).

o Вывести на экран условие задачи и данные исполнителя. Реализовать вывод этой информации с помощью отдельной функции. 9

o Вычисление результата на основании исходных данных реализовать в отдельной функции.

o В тексте каждой функции пояснить ее назначение в комментарии

o В функции main() организовать ввод данных, вызов функции, вывод результатов.

o В функции main() в комментарии указать данные исполнителя

o Ввод и вывод данных оформить соответствующими пояснениями с указанием единиц измерения

o Проверить работу программы на 1 тестовом примере (ответ легко подсчитать устно), подобранном самостоятельно.

o С помощью программы выполнить расчет для 3 различных корректных наборов данных, подобранных самостоятельно.

ЗАДАЧА 2. Дополнить программу решением следующей задачи (Варианты приведены в Приложении 2).

o Вывести на экран условие варианта задачи. Реализовать вывод этой информации с помощью отдельной функции.

o Вычисление всех результатов на основании исходных данных реализовать в одной отдельной функции.

o В тексте каждой функции пояснить ее назначение в комментарии o В функции main() организовать ввод данных, вызов функции, вывод результатов.

o В функции main() в комментарии указать данные исполнителя

o Ввод и вывод данных оформить соответствующими пояснениями с указанием единиц измерения

o Проверить работу программы на 1 тестовом примере (ответ легко подсчитать устно), подобранном самостоятельно.

o С помощью программы выполнить расчет для 3 различных корректных наборов данных, подобранных самостоятельно.

ЗАДАЧА 1. Создать новый проект и повторить все примеры, добавляя их (с учетом уже набранных команд) в одну программу.

Изменить численные константы в тексте программы и выполнить программу 3 раза с различными входными наборами данных.

Результаты включить в отчет.

using System;

namespace ConsoleApplication6;

class Program

{

public static double AreaCount(double a, double b)

//--вычисление площади прямоугольника

{

double s = a \* b;

return s;

}

public static double DiagonalCount(double a, double b)

{

double c = Math.Sqrt(a \* a + b \* b);

return c;

}

public static void Info() //--заголовок на экране

{

Console.WriteLine("=== Расчет площади и диагонали прямоугольник ===");

}

public static void Output(double a, double b, double s, double c)

{

//--вывод рузельтатов вычислений

Console.WriteLine("Результат вычислений:");

Console.WriteLine("Для прямоугольника со сторонами a={0,6:F3}, b={1,6:F3}", a, b);

Console.WriteLine("Площадь s={0,7:F3}", s);

Console.WriteLine("Диагональ c={0,6:F3}", c);

Console.WriteLine("-------------------------------------------------------");

}

public static void Input(out double a, out double b)

//--ввод исходных данных

{

Console.Write("Введите длинну первой стороны прямоугольника: ");

a = double.Parse(Console.ReadLine());

Console.Write("Введите длинну второй стороны прямоугольника: ");

b = double.Parse(Console.ReadLine());

Console.WriteLine("Для пряиоугольника со сторонами a={0,6:F3}, b={1,6:F3}", a, b);

}

public static void Rectangle(double a, double b, out double s, out double c)

{

s= a \* b;

c = Math.Sqrt(a \* a + b \* b);

}

static void swap(ref int a,ref int b)

{//--Статический метод для обмена значениями//аргументы

int t = b;

b = a;

a = t;

}

static void Main(string[] args)

{

Info();

double side1, side2, area, diagonal;

Input(out side1, out side2);

area = AreaCount(side1, side2);

diagonal = DiagonalCount(side1, side2);

Output(side1, side2, area, diagonal);

Rectangle(side1, side2, out area, out diagonal);

Output(side1, side2, area, diagonal);

int u=10,v=200;

Console.WriteLine("u={0} v={1}",u,v);

swap(ref u, ref v);

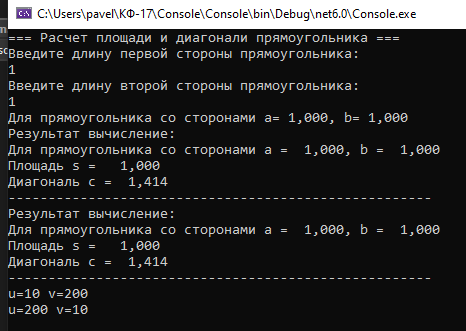
Console.WriteLine("u={0} v={1}", u, v);

Console.ReadLine();

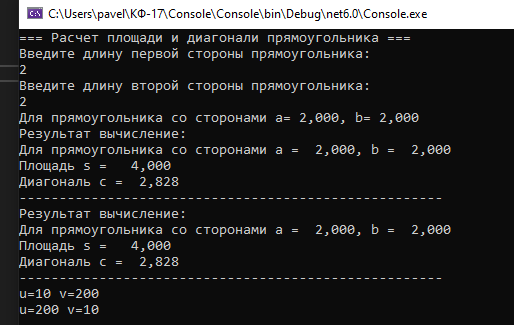
}

}

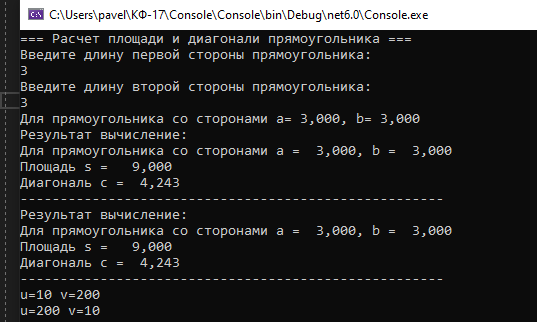
При a=1; и b=1;



При a=2; и b=2;



При a=3; и b=3;



ПРИЛОЖЕНИЕ 1. Варианты задания к Задаче 1.

Вычислить давление n молей идеального газа, занимающего объем V при температуре T.

using System;

namespace ConsoleApplication6

{

class Progran

{

public static void Main()

{

static double Davlenie(double n, double V, double T)

// вычисление давления

{

double R = 8.31;

double p = n \* R \* T / V;

return p;

}

Console.WriteLine("Вычислить давление n молей идеального газа, занимающего объем V ,при температуре T");

double n, V, T, p;

Console.Write("Введите значение n=");

n = double.Parse(Console.ReadLine());

Console.Write("Введите значение V=");

V = double.Parse(Console.ReadLine());

Console.Write("Введите значение T=");

T = double.Parse(Console.ReadLine());

p = Davlenie(n, V, T);

Console.Write("При кол-во n={0} (моль),значении V={1} (м^3),значении T={2} (C)", n, V, T);

Console.WriteLine("Значение p={0}(Па)", p);

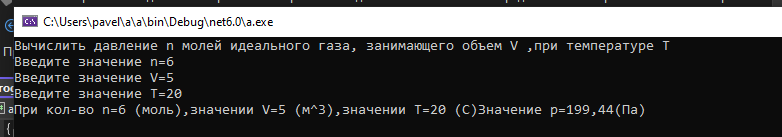
Console.ReadLine();

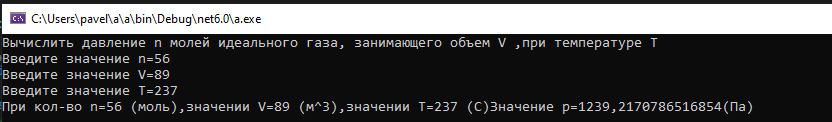
}

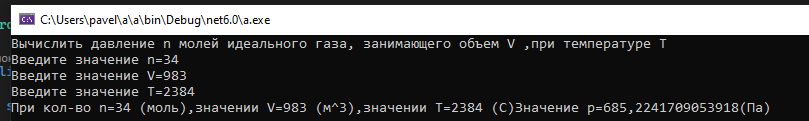
}

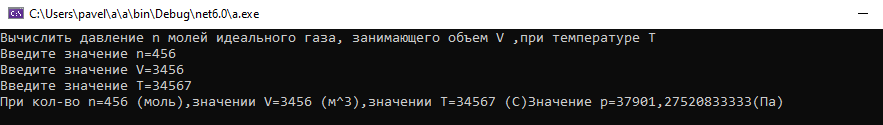
}

Результат выполнения задачи при вводных n=6(молей) ; V=5(); T=20 ();









ПРИЛОЖЕНИЕ 2. Варианты задания к Задаче 2.

Реализовать вычисление периода и частоты колебаний в колебательном контуре, образованном емкостью C и индуктивностью L

using System;

namespace ConsoleApplication6

{

class Progran

{

public static void fun(double L, double C)

// вычисление период

{

double T = 2 \* Math.PI \* Math.Sqrt(L \* C);

double v = 1 / (2 \* Math.PI \* Math.Sqrt(L \* C));

}

public static void Main()

{

Console.WriteLine("Реализовать вычисление периода и частоты колебаний в колебательном контуре, образованном емкостью C и индуктивностью L");

double L, C, t, c;

Console.Write("Введите значение L=");

L = double.Parse(Console.ReadLine());

Console.Write("Введите значение C=");

C = double.Parse(Console.ReadLine());

fun(L,C);

Console.WriteLine("Значение T={0:f3}(С) \n Значение V={0:f24}(Гц)");

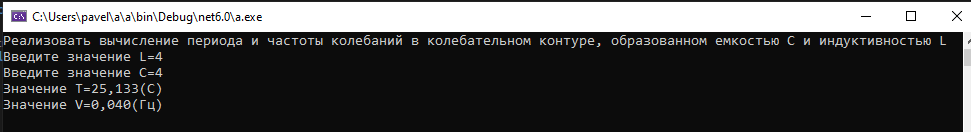
Console.ReadLine();

}

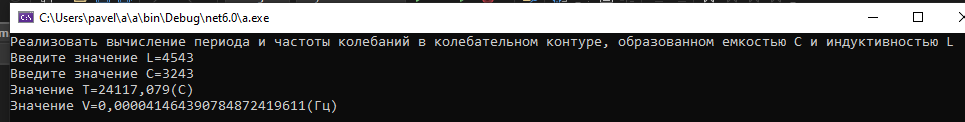
}

}

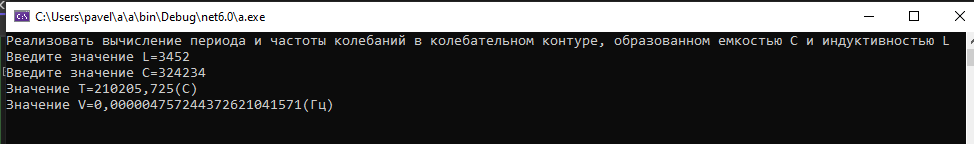
Результат выполнения задачи при вводных L=4(Гн); C=4(Ф);



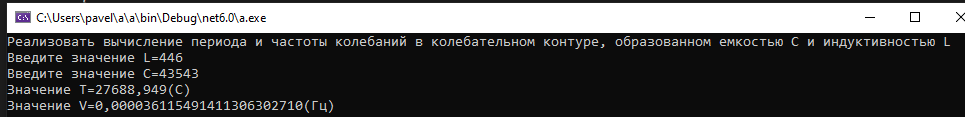
1.



2.



3.



**КОНТРОЛЬНЫЕ ВОПРОСЫ**

1. Назначение методов в программе

Метод класса - это именованный блок выполняемого кода (набор операторов), который может быть вызван на выполнение из разных частей программы. При вызове метода он выполняет свой код, а затем возвращает управление в точку вызова.

1. Общая структура методов в языке С#

Модификатор ТипРезультата ИмяМетода(тип имя, тип имя, . . )

{ . . . выполняемые операторы

}

1. Локальные переменные методов

Примеры заголовков методов:

void Print(int p, double mass) {...}

private double Triangle(double a, double b, double c){...}

public static void Count(double z, out int k){...}

1. Формальные и фактические параметры методов

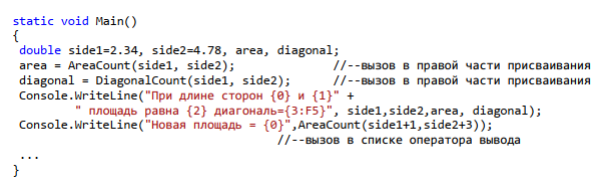
Между формальными и фактическими параметрами должно быть соответствие по количеству, порядку следования, типу и статусу параметров.

1. Передача параметров по значению

Передачей параметров по значению и состоит в том, что на самом деле в метод передается копия записанного в точке вызова фактического параметра. Это означает, что значение передаваемой переменной невозможно изменить внутри метода. Как только метод завершил свою работу, все локальные переменные, в том числе и копии переменных-аргументов, автоматически уничтожаются.

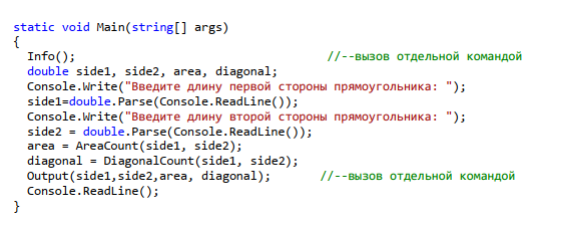
1. Передача параметров по адресу
2. Оформление и вызов методов, вычисляющих объявленный результат

По сути, метод вызывается как обычная встроенная функция Math.Sin(), Math.Cos() и другие. Для таких методов используют название методфункция.

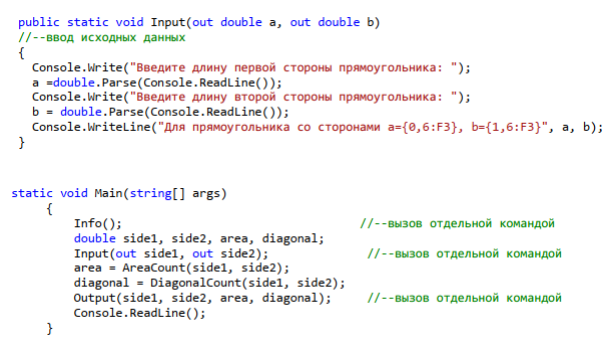


Метод обязательно должен содержать команду return ИмяРезультата.

1. Оформление и вызов методов, не вычисляющих объявленный результат



1. Оформление и вызов методов с выходными параметрами



1. Оформление и вызов методов с изменяемыми параметрами

