|  |
| --- |
| Пермский филиал федерального государственного автономного образовательного учреждения высшего образования  «Национальный исследовательский университет  «Высшая школа экономики»  *Факультет экономики, менеджмента и бизнес-информатики* |
|  |
| Меньшиков Олег Владимирович  по направлению подготовки *09.03.04 Программная инженерия*  образовательная программа «Программная инженерия»   |  |  | | --- | --- | |  | Доцент к[афедры информационных технологий в бизнесе](https://perm.hse.ru/bi/infobis/), кандидат технических наук  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  О.Л. Викентьева | |

Пермь, 2020 год

Содержание

[*1* Постановка задачи 3](#_Toc52012517)

[2 Формулы, используемые при решении задачи 3](#_Toc52012518)

[3 Проектирование 3](#_Toc52012519)

[Задача 1 3](#_Toc52012520)

[Задача 2 5](#_Toc52012521)

[Задача 3 6](#_Toc52012522)

[4 Программы для решения задач на языке C# 6](#_Toc52012524)

[Задача 1 6](#_Toc52012525)

[Задача 2 8](#_Toc52012526)

[Задача 3 8](#_Toc52012527)

[5 Описание используемых в программе методов класса Math 9](#_Toc52012528)

[6. Объяснение результатов работы программы 9](#_Toc52012529)

[Задача 1 9](#_Toc52012530)

[Задача 2 9](#_Toc52012531)

[Задача 3 10](#_Toc52012532)

# **Постановка задачи**

* 1. Для задачи 1 определить тип заданных выражений и найти их значения.
  2. Составить систему тестов и вычислить полученное выражение для нескольких значений Х, определить при каких Х выражение не может быть вычислено.
  3. Для задачи 2 записать выражение, зависящее от координат точки X1 и Y1 и принимающее значение TRUE, если точка принадлежит заштрихованной области, и FALSE, если не принадлежит.
  4. Составить систему тестов и вычислить полученное выражение для нескольких точек, принадлежащих и не принадлежащих заштрихованной области.
  5. Для задачи 3 вычислить значение выражения, используя различные вещественные типы данных (float и double).
  6. Результаты всех вычислений вывести на печать.
  7. Объяснить полученные результаты.

# **Формулы, используемые при решении задачи**

* 1. В программе использовалась формула функции прямых на плоскости: х=0, у=0, у= –1/7\*х – 1.

# **Проектирование**

## Задача 1

Начало

m

TryParse

-

+

n

TryParse

-

+

+

«Невозможно посчитать»

If n=1

-

res1 = m / (n-1)

n=n-1

If n=0

+

«Невозможно посчитать»

-

res2 = m / n < n

n=n-1

res3 = m + n > n + m; n=n+1

x

TryParse

-

+

Res 4 = X^2\* |x – 1|^1/2 + |25-x^5|

***Рисунок 3.1 ­­­­­­­— Схема задачи 1***

## Задача 2

Начало

х

TryParse

-

+

y

TryParse

-

+

-

«Не принадлежит»

If oblast = true

+

«Принадлежит»

***Рисунок 3.1 ­­­­­­­— Схема задачи 1***

## Задача 3

# 

Начало

a=1000, b=0,0001;

c = (a+b)^2;

d = a^3;

e = 3 \* a^2 \* b;

f = 3 \* a \* b^2;

g = b^3;

res = (c - (d + e)) / (f + g);

***Рисунок 3.3 – Схема задачи 3***

Вывод res

# **Программы для решения задач на языке C#**

## Задача 1

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace задача1.\_1

{

class Program

{

static void Main(string[] args)

{

int m = 0, n = 0, res1 = 0;

bool ok, res2, res3;

string buf;

double x = 0, res4 = 0;

do

{

Console.WriteLine(“Введите число целое m:”);

buf = Console.ReadLine();

ok = int.TryParse(buf, out m);

if (ok == false) Console.WriteLine(«Введено ошибочное число, введите число заново»);

}

while (ok == false);

do

{

Console.WriteLine(“Введите целое число n:”);

buf = Console.ReadLine();

ok = int.TryParse(buf, out n);

if (!ok) Console.WriteLine(«Введено ошибочное число, введите заново»);

}

while (!ok);

if (n == 1) Console.WriteLine(“Операция невозможна”);

else

{

res1 = m / (--n);

n++;

Console.WriteLine($”m={m}, n={n}, m / (--n++) = {res1}”);

}

if (n == 0) Console.WriteLine(«Операция невозможна»);

else

{

res2 = m / n < n--;

Console.WriteLine($”m={m}, n={n}, m / n < n-- = {res2}”);

}

res3 = m + n++ > n + m;

Console.WriteLine($”m={m}, n={n}, m + n++ > n + m = {res3}”);

do

{

Console.WriteLine(«введите число х вещественного типа:»);

buf = Console.ReadLine();

ok = double.TryParse(buf, out x);

if (ok == false) Console.WriteLine(“Введено ошибочное число, введите снова”);

}

while (ok == false);

res4 = (Math.Pow(x, 2) \* Math.Sqrt(Math.Abs(x – 1))) + Math.Abs(25 – Math.Pow(x, 5));

Console.WriteLine(«Полученный результат = {0}», res4);

}

}

}

## Задача 2

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Задача2

{

class Program

{

static void Main(string[] args)

{

double x=0, y=0;

bool ok, oblast;

string buf;

do

{

Console.WriteLine(“Введите координату х:”);

buf = Console.ReadLine();

ok = double.TryParse(buf, out x);

if (ok == false) Console.WriteLine(«Ошибочное число, введите заново»);

}

while (ok == false);

do

{

Console.WriteLine(“Введите число y:”);

buf = Console.ReadLine();

ok = double.TryParse(buf, out x);

if (ok == false) Console.WriteLine(«Введено ошибочное число, введите заново»);

}

while (ok == false);

oblast = (x<=0) && (y<=0) && (y >= (-1/7)\*x -1);

Console.WriteLine($”x={x}, y={y}, x<=0) && (y<=0) && (y >= (-1/7)\*x -1 = {oblast}”); ;

}

}

}

## Задача 3

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace задача3

{

class Program

{

static void Main(string[] args)

{

double a = 1000, b = 0.0001;

double c, d, e, f, g, res;

c = Math.Pow(a + b, 3);

d = Math.Pow(a, 3);

e = 3 \* a \* a \* b;

f = 3 \* a \* b \* b;

g = Math.Pow(b, 3);

res = (c – (d + e)) / (f + g);

Console.WriteLine(“тип double даст результат = {0}”, res);

float af = 1000, bf = 0.0001f;

float cf, df, ef, ff, gf, resf;

cf = (float)Math.Pow(af + bf, 3);

df = (float)Math.Pow(af, 3);

ef = (float)3 \* af \* af \* bf;

ff = (float)3 \* af \* bf \* bf;

gf = (float)Math.Pow(bf, 3);

resf = (float)(cf – (df + ef)) / (ff + gf);

Console.WriteLine(“тип float даст результат = {0}”, resf);

}

}

}

# **Описание используемых в программе методов класса Math**

|  |  |  |
| --- | --- | --- |
| Функция | Описание | Пример вызова |
| Abs() | Абсолютное значение | int a=Math.Abs(-10); |
| Pow() | Возведение в степень | double x=Math.Pow(2,3); |
| Sqrt() | Квадратный корень | double x; . . . double y=Math.Sqrt(x); |

Таблица 1 – Описание методов

# **Объяснение результатов работы программы**

## Задача 1

В задаче 1 нам были предложены математические выражения и неравенства. Работа велась в целых числах, поэтому результаты точны. Но были моменты, в которых программа могла сломаться: вводятся числа иного типа, вводятся числа, которые приведут к делению на 0. Эти моменты предусмотрены в представленной выше программе.

## Задача 2

В этой задаче было необходимо определить принадлежность/непринадлежность точки к заданной области. Для решения этой задачи потребовались уравнения графиков функций. Использовалась переменная булевого типа, которой присваивалось логическое выражение, состоящее из уравнений графиков функций.

## Задача 3

С первого взгляда программа разделена на абсолютно одинаковые части. Но программа выведет 2 разных значения, ибо в первом и во втором случае мы используем переменные разного типа: double и float, которые имеют различную точность. Из этого мы делаем вывод, что при выборе типа переменных будет зависеть точность результатов, полученных программой.