ioПермский Национальный Исследовательский Политехнический Университет

Электротехнический факультет

Кафедра «Информационные технологии и автоматизированные системы»

Лабораторная работа №1

по дисциплине

«Объектно-ориентированное программирование»

Выполнил:

студент группы ИВТ-19-2б

Шеретов Марк

Проверила:

доцент кафедры ИТАС

Викентьева О. Л.

Пермь 2020

**Содержание**

1. Постановка задачи 3

2. Математическая модель 3

3. Программы на C# 5

4. Описание используемых методов класса Math 8

5. Система тестов для проверки правильности работы программы и результаты выполнения тестов 8

5.1 Таблицы тестов 9

5.2 Проверка достаточности тестов (по критериям черного ящика) 10

6. Объяснение результатов работы программы 11

1. **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.

**Вариант 19**

|  |  |  |
| --- | --- | --- |
| Задача 1 | Задача 2 | Задача 3 |
| 1) (n++/--m)++ 2) ++m<n— 3)--m>++n  4) |  | a=1000, b=0.0001 |

**2. Математическая модель**

**Задача 1:** Нахождение области допустимых значений в 4 выражении:

F = .

Арктангенс пределяется как функция, обратная к косинусу, на всей числовой прямой ().

Дополнительная обработка входных значений не требуется.

**Задача 2:**Программа определяет вхождение заданной пользователем точки в ограниченную графиками область.

Сначала нужно проверить вхождение точки в треугольник, расположённый справа от оси ординат. Для этого разделим его на 3 линии и составим их уравнения:

1. x = 0,
2. y = -x + 5,
3. y = x – 5.

Если точка входит в ограниченную этими прямыми область, то булевая переменная f1 = true, иначе false. Для проверки используется условие:

(x >= 0) && ((-x + 5) >= y) && ((x - 5) <= y)

Далее нужно проверить вхождение точки в область маленького треугольника, лежащего слева от оси ординат. Для этого используются уравнения прямых:

1. x = 0,
2. y = 0,

3) y = -x -5.

Если точка входит в область, ограниченную окружностью, то переменная f2 = true, иначе false. Для проверки используется условие:

(x <= 0) && (-x - 5 <= y) && (y <=0)

Если точка входит в одну из областей или в обе, то переменная isInhere становится true. Если не входит ни в одну, то false.

**3. Программы на языке C#**

**Задача 1**

using System;

namespace Z1

{

class Z1

{

private static int n, m;

static void Main(string[] args)

{

string buf;

bool ok;

do

{

Console.Write("n? ");

buf = Console.ReadLine();

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

if (!ok) Console.WriteLine("Введите целое число");

} while (!ok);

do

{

Console.Write("m? ");

buf = Console.ReadLine();

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

if (!ok) Console.WriteLine("Введите целое число");

if(m == 1) {Console.WriteLine("Ошибка.m = 1 недопустимо!"); ok = false;}

} while (!ok);

Func1();

Func2();

Func3();

Func4();

}

static void Func1()

{

int res1 = (n++ / --m);

res1++;

Console.WriteLine($"(n++/--m)++ = {res1}, после операции: n = {n}, m = {m}");

NextCommand();

}

static void Func2()

{

bool res2 = ++m < n--;

Console.WriteLine($"++m<n-- = {res2}, после операции: n = {n}, m = {m}");

NextCommand();

}

static void Func3()

{

bool res3 = --m > ++n;

Console.WriteLine($"--m > ++n = {res3}, после операции: n = {n}, m = {m}");

NextCommand();

}

static void Func4()

{

bool ok;

string buf;

double x;

do

{

Console.Write("x? ");

buf = Console.ReadLine();

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

if (!ok) Console.WriteLine("Введите число");

} while (!ok);

double k = Math.Pow(x,2);

float res4 = 7 \* (float)Math.Atan(k);

Console.WriteLine($"arctan(x^2) = {res4}");

NextCommand();

}

static void NextCommand()

{

Console.WriteLine("Press any key to continue");

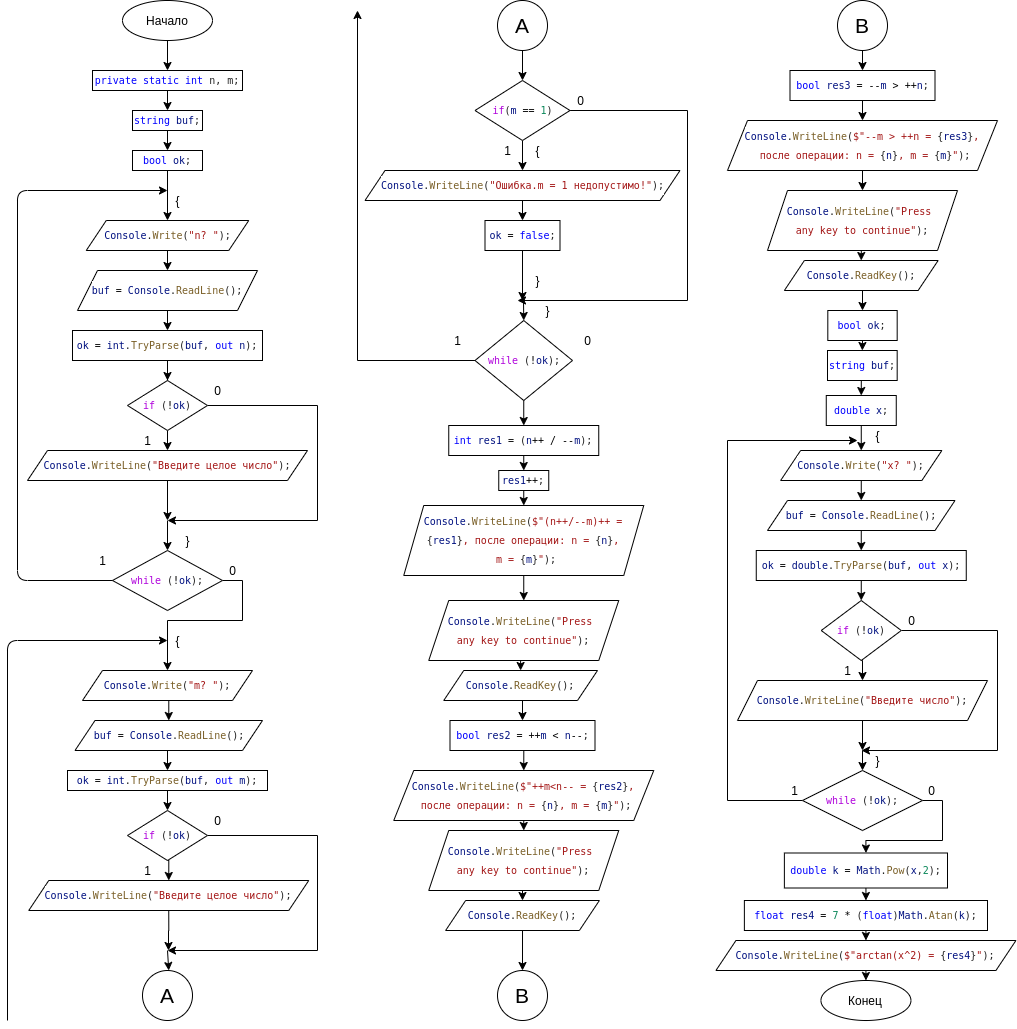
Console.ReadKey();

}

}

}

**блок-схема**



**Задача 2**

using System;

namespace Z2

{

class Z2

{

static void Main(string[] args)

{

string buf;

bool ok;

double x, y;

do

{

Console.Write("x? ");

buf = Console.ReadLine();

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

if (!ok) Console.WriteLine("Введите число");

} while (!ok);

do

{

Console.Write("y? ");

buf = Console.ReadLine();

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

if (!ok) Console.WriteLine("Введите число");

} while (!ok);

bool f1 = (x >= 0) && ((-x + 5) >= y) && ((x - 5) <= y);

bool f2 = (x <= 0) && (-x - 5 <= y) && (y <=0);

bool isInhere = (f1) || (f2);

Console.WriteLine("Большой треугольник - " + f1);

Console.WriteLine("Малый треугольник - " + f2);

Console.WriteLine("Входит в область - " + isInhere);

}

}

}

**Задача 3**

using System;

namespace Z3

{

class Z3

{

static void Main(string[] args)

{

Func1();

Func2();

}

static void Func1()

{

int a = 1000;

float b = 0.0001f;

float r1 = (float)Math.Pow(a + b, 2);

float r2 = (float)Math.Pow(a,2)+ 2\*a\*b;

float r3 = (float)Math.Pow(b,2);

float r4 = r1 - r2;

float r = r4/r3;

Console.WriteLine("Тип float - " + r);

}

static void Func2()

{

int a = 1000;

double b = 0.0001;

double r1 = Math.Pow(a + b, 2);

double r2 = Math.Pow(a,2)+ 2\*a\*b;

double r3 = Math.Pow(b,2);

double r4 = r1 - r2;

double r = r4/r3;

Console.WriteLine("Тип double - " + r);

}

}

}

**4. Описание используемых методов класса Math**

Класс Math используется для вычисления значений множества математических функций и операций.

1. double Math.Pow(double X, double Y) – используется для возведения числа X в степень Y. Возвращает значение типа double.
2. double Math.Atan(double X) – используется для получения значения функции arctg от значения X. Возвращает значение радианов типа double.

**5. Система тестов для проверки правильности работы программы и результаты выполнения тестов**

**Задача 1**

Значения n может быть любым целым числом. Необходимо проверить ввод отрицательных/положительных чисел, нуля, символов, вещественных чисел.

Значения m любое целое число кроме 1. Необходимо проверить ввод отрицательных/положительных чисел, нуля, единицы, вещественных чисел, символов.

Значения x могут быть любыми целыми и вещественными числами. Необходимо проверить ввод отрицательных/положительных чисел, нуля, символов, вещественных чисел.

**Задача 2**

Значения x и y могут попадать в область одного из графиков функций, в обе области или ни в одну.

Необходимо проверить ввод значений типа int и double, точек попадающих в область и не попадающих соответственно.

**5.1 Таблицы тестов**

**Задача 1**

|  |  |  |  |
| --- | --- | --- | --- |
| **Значения переменных** | **Ожидаемый результат** | **Полученный результат** | **Номер теста** |
| n = 0, m = 1 | Ошибка, m = 1 недопустимо | + | 1 |
| n = -5, m = odin | Ошибка, введите целое число | + | 2 |
| n = 10,25, m = 0 | Ошибка, введите целое число | + | 3 |

|  |  |  |  |
| --- | --- | --- | --- |
| n = 120, m = -120 | result = 1, n = 121, m = -121  result = true, n = 120, m = -120  result = false, n = 121, m = -121 | + | 4 |
| x = -5 | result =10,715723 | + | 5 |
| x = -0,75 | result = 3,5867264 | + | 6 |
| x = 0 | result = 0 | + | 7 |
| x = 2 | result = 9,280724 | + | 8 |
| x = 0,25 | result = 0,43693167 | + | 9 |
| x = asd | Ошибка, введите число | + | 10 |

**Задача 2**

|  |  |  |  |
| --- | --- | --- | --- |
| **Значения переменных** | **Ожидаемый результат** | **Полученный результат** | **Номер теста** |
| x = 0, y = 0 | Большой треугольник - True  Малый треугольник - True  Входит в область - True | + | 1 |
| x = 5,75, y = -2,6 | Большой треугольник - False  Малый треугольник - False  Входит в область - False | + | 2 |
| x = -5, y = 0 | Большой треугольник - False  Малый треугольник - True  Входит в область - True | + | 3 |
| x = 1, y = 4,45 | Большой треугольник - False  Малый окружность - False  Входит в область - False | + | 4 |
| x = asd, y = 5 | Ошибка, введите число | + | 5 |

**5.2 Проверка достаточности тестов (по критериям черного ящика)**

**Задача 1**

**Входные данные**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Тесты** | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| n – int,  m – int | + |  |  |  |  |  |  |  |  |  |
| n – int < 0,  m – string |  | + |  |  |  |  |  |  |  |  |
| n – double,  m – int |  |  | + |  |  |  |  |  |  |  |
| n – int > 0,  m – int < 0 |  |  |  | + |  |  |  |  |  |  |
| x < -1,  ()>1 |  |  |  |  | + |  |  |  |  |  |
| -1 <= x <=1,  (x2)<=1 |  |  |  |  |  | + |  |  |  |  |
| x = 0,  (x2)<=1 |  |  |  |  |  |  | + |  |  |  |
| x > 1,  (x2)>1 |  |  |  |  |  |  |  | + |  |  |
| 0 < x <= 1,  0 < (x2) <=1 |  |  |  |  |  |  |  |  | + |  |
| x = asd |  |  |  |  |  |  |  |  |  | + |

**Выходные данные**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Ошибка |  | + | + |  | + |  |  | + |  | + |
| Число | + |  |  | + |  | + | + |  | + |  |

**Задача 2**

**Входные данные**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Тесты** | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 |
| x – входит в оба,  y – входит в оба | + |  |  |  |  |  |  |  |  |  |  |  |  |
| x – никуда не вх.,  y – никуда не вх. |  | + |  |  |  |  |  |  |  |  |  |  |  |
| x – входит в малый тр.  y – входит в малый тр. |  |  | + |  |  |  |  |  |  |  |  |  |  |
| x - входит в большой тр., y - никуда не вх. |  |  |  | + |  |  |  |  |  |  |  |  |  |
| x – строка,  y – int |  |  |  |  | + |  |  |  |  |  |  |  |  |
| x - входит в малый тр. левая граница  y - входит в малый тр. Верхняя граница |  |  |  |  |  | + |  |  |  |  |  |  |  |
| x - входит в больш. тр. Левая граница,  y - входит в больш. тр. Верхняя граница |  |  |  |  |  |  | + |  |  |  |  |  |  |
| x - входит в больш. тр. Правая граница,  y - входит в больш. тр. Середина. |  |  |  |  |  |  |  | + |  |  |  |  |  |
| x - входит в больш. тр. Левая граница,  y - входит в больш.тр. Нижняя граница. |  |  |  |  |  |  |  |  | + |  |  |  |  |
| x - входит в боль. тр. Правое верхнее ребро,  y - входит в боль. тр.  Правое верхнее ребро |  |  |  |  |  |  |  |  |  | + |  |  |  |
| входит в боль. тр. Правое нижнее ребро,  y - входит в боль. тр.  Правое нижнее ребро. |  |  |  |  |  |  |  |  |  |  | + |  |  |
| входит в мал. тр. лев нижнее ребро,  y - входит в мал. тр.  лев нижнее ребро. |  |  |  |  |  |  |  |  |  |  |  | + |  |
| входит в мал. тр. верхнее ребро,  y - входит в мал. тр.  верхнее ребро. |  |  |  |  |  |  |  |  |  |  |  |  | + |

**Выходные данные**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| True | + |  | + |  |  | + | + | + | + | + | + | + | + |
| False |  | + |  | + |  |  |  |  |  |  |  |  |  |
| Ошибка |  |  |  |  | + |  |  |  |  |  |  |  |  |

**6. Объяснение результатов работы программы**

В Задаче 3 результаты различаются, так как используется типы данных с разной точностью.