**Лабораторная работа №13**

**РАЗРАБОТКА ПОДПРОГРАММ: РЕАЛИЗАЦИЯ ПРЕДИКАТОВ И МАТЕМАТИЧЕСКИХ ФУНКЦИЙ**

**Цели:**

1. Получить навыки разработки простейших функций.
2. Научиться записывать сложные условия оператора ветвления в виде функций-предикатов.

**Приложение Lab13\_01. Разработка приложения, определяющего попадает ли точка в заданную область**

***Задание*:** Разработать консольное приложение, организующее ввод координат точки и проверяющее, принадлежит ли эта точка заштрихованной области (рисунок 1.0), включая ее границу. В результате работы должно текстовое сообщение «Точка принадлежит области» либо «Точка не принадлежит области».

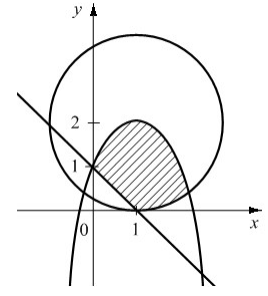


Рисунок 1.0

***Константы:*** отсутствуют.

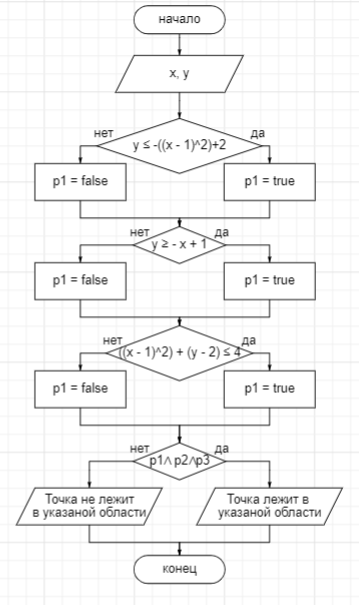
***Переменные:*** p1, p2, p3 – переменные типа bool; x, y – координаты точки, переменные типа double;

***Исходные данные:*** x, y;

***Результат:***

Программа определяет принадлежит ли точка заштрихованной области.

***Блок-схема:***



***Код приложения:***

**Листинг 1 – Код консольного приложения Lab13\_01**

namespace PuzanovVE.OP.Lab13\_01

{

class Program

{

static void Main(string[] args)

{

//Стандартная шапка

Console.WriteLine("Лабораторная работа №13");

Console.WriteLine("Приложение Lab13\_01");

Console.WriteLine("Разработчик: Пузанов В. Е.");

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

//информация о задаче

Console.WriteLine("Разработка приложения, определяющего попадает ли точка в заданную область");

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

//основной код программы

// парабола y <= - (x-1)^2 + 2

// прямая y >= - x + 1

// круг (x - 1 )^2 + (y - 2) <= 4

bool p1, p2, p3;

double x, y;

Console.WriteLine("Введите x");

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

Console.WriteLine("Введите y");

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

if (y <= -Math.Pow((x - 1), 2) + 2)

p1 = true;

else

p1 = false;

if (y >= - x + 1)

p2 = true;

else

p2 = false;

if (Math.Pow((x - 1), 2) + (y - 2) <= 4)

p3 = true;

else

p3 = false;

if (p1 && p2 && p3)

Console.WriteLine("Точка лежит в указанной области");

else

Console.WriteLine("Точка не лежит в указанной области");

Console.ReadLine();

}

}

}

**Тестирование:**

Контрольный пример 1:

Исходные данные:

x = 5; y = 3;

Результат:

Точка не лежит в указанной области

Результат работы программы для указанных исходных данных приведён на рисунке 1.1

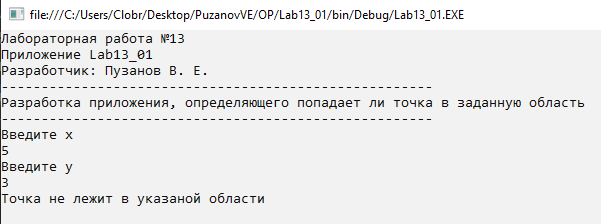


Рисунок 1.1 –Результат работы программы Lab13\_01

Контрольный пример 2:

Исходные данные:

x = 1; y = 1;

Результат:

Точка лежит в указанной области

Результат работы программы для указанных исходных данных приведён на рисунке 1.2

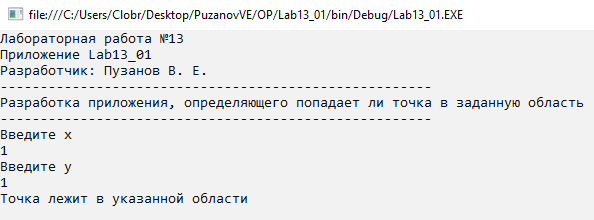


Рисунок 1.2 –Результат работы программы Lab13\_01

Контрольный пример 3:

Исходные данные:

x = 0,1; y = -1;

Результат:

Точка не лежит в указанной области

Результат работы программы для указанных исходных данных приведён на рисунке 1.3

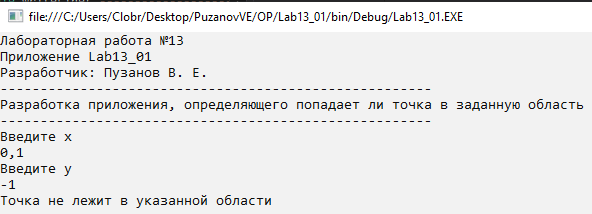


Рисунок 1.3 –Результат работы программы Lab13\_01

**Приложение Lab13\_02. Разработка приложения, вычисляющего факториал числа**

***Задание*:** Разработать консольное приложение, вычисляющее факториал целого неотрицательного числа n , определяемый согласно формуле:

***Константы:*** отсутствуют.

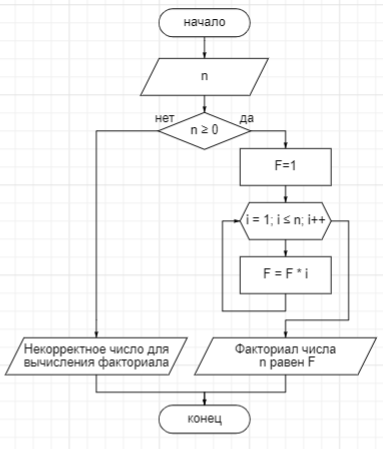
***Переменные:*** F – переменная хранящая в себе значение интеграла; n – вводимое число, интеграл которого необходимо найти.

***Исходные данные:*** n;

***Результат:***

Программа вычисляет интеграл вводимого числа.

***Блок-схема:***



***Код приложения:***

**Листинг 2 – Код консольного приложения Lab13\_02**

namespace PuzanovVE.OP.Lab13\_02

{

class Program

{

static void Main(string[] args)

{

//Стандартная шапка

Console.WriteLine("Лабораторная работа №13");

Console.WriteLine("Приложение Lab13\_02");

Console.WriteLine("Разработчик: Пузанов В. Е.");

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

//информация о задаче

Console.WriteLine("Разработка приложения, вычисляющего факториал числа");

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

//основной код программы

int F;

Console.WriteLine("Введите число, из которого нужно вычислить интеграл:");

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

if (n >= 0)

{

F = 1;

for (int i = 1; i <= n; i++)

{

F = F \* i;

}

Console.WriteLine("Факториал числа " + n + " равен " + F);

}

else

{

Console.WriteLine("Некорректное число для вычисления факториала");

}

Console.ReadLine();

}

}

}

**Тестирование:**

Контрольный пример 1:

Исходные данные:

n = 5;

Результат:

Факториал числа 5 равен 120

Результат работы программы для указанных исходных данных приведён на рисунке 2.1

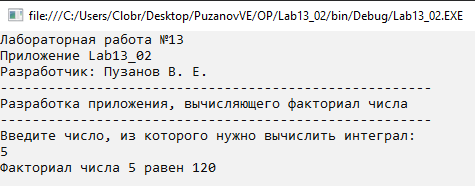


Рисунок 2.1 –Результат работы программы Lab13\_02

Контрольный пример 2:

Исходные данные:

n = -5;

Результат:

Некорректное число для вычисления факториала

Результат работы программы для указанных исходных данных приведён на рисунке 2.2

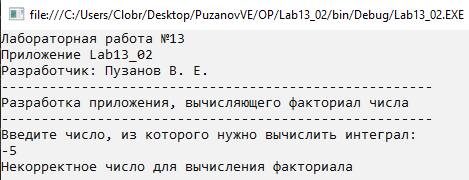


Рисунок 2.2 –Результат работы программы Lab13\_02

Контрольный пример 3:

Исходные данные:

n = 0;

Результат:

Факториал числа 0 равен 1

Результат работы программы для указанных исходных данных приведён на рисунке 2.3

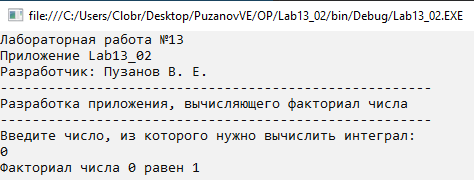


Рисунок 2.3 –Результат работы программы Lab13\_02

**Приложение Lab13\_03. Разработка функции, определяющей попадает ли точка в заданную область**

***Задание*:** Разработать функцию, реализующую предикат, принимающий значение «истина», если точка с координатами (x; y) принадлежит заштрихованной области (рисунок 1.0), включая ее границу, и значение «ложь», если не принадлежит. Разработать консольное приложение, использующее эту функцию.

***Константы:*** отсутствуют.

***Переменные:*** p1, p2, p3 – переменные типа bool; x, y – координаты точки, переменные типа double;

***Исходные данные:*** x, y;

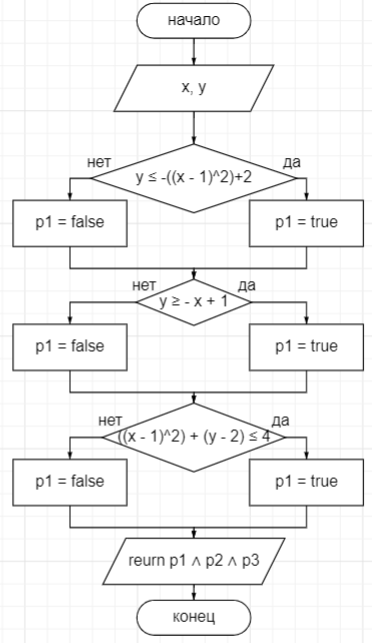
***Результат:***

Программа определяет принадлежит ли точка заштрихованной области.

***Блок-схема:***



***Блок-схема функции Predicate (x, y):***



***Код приложения:***

**Листинг 3 – Код консольного приложения Lab13\_03**

namespace PuzanovVE.OP.Lab13\_03

{

class Program

{

static bool Predicate (double x, double y)

{

bool p1, p2, p3;

if (y <= -Math.Pow((x - 1), 2) + 2)

{

p1 = true;

}

else

{

p1 = false;

}

if (y >= -x + 1)

{

p2 = true;

}

else

{

p2 = false;

}

if (Math.Pow((x - 1), 2) + (y - 2) <= 4)

{

p3 = true;

}

else

{

p3 = false;

}

return (p1 && p2 && p3);

}

static void Main(string[] args)

{

//Стандартная шапка

Console.WriteLine("Лабораторная работа №13");

Console.WriteLine("Приложение Lab13\_03");

Console.WriteLine("Разработчик: Пузанов В. Е.");

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

//информация о задаче

Console.WriteLine("Разработка приложения, определяющего попадает ли точка в заданную область");

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

//основной код программы

double x, y;

Console.WriteLine("Введите x");

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

Console.WriteLine("Введите y");

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

if (Predicate(x, y))

{

Console.WriteLine("Точка лежит в указаной области");

}

else

{

Console.WriteLine("Точка не лежит в указаной области");

}

Console.ReadLine();

}

}

}

**Тестирование:**

Контрольный пример 1:

Исходные данные:

x = 5; y = 3;

Результат:

Точка не лежит в указанной области

Результат работы программы для указанных исходных данных приведён на рисунке 3.1

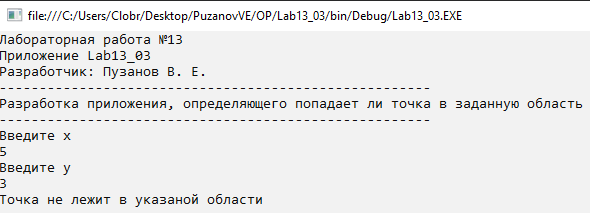


Рисунок 3.1 –Результат работы программы Lab13\_03

Контрольный пример 2:

Исходные данные:

x = 1; y = 1;

Результат:

Точка лежит в указанной области

Результат работы программы для указанных исходных данных приведён на рисунке 3.2

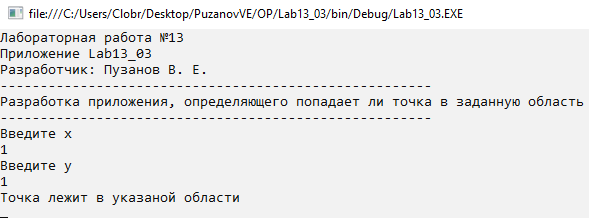


Рисунок 3.2 –Результат работы программы Lab13\_03

Контрольный пример 3:

Исходные данные:

x = 0,1; y = -1;

Результат:

Точка не лежит в указанной области

Результат работы программы для указанных исходных данных приведён на рисунке 3.3

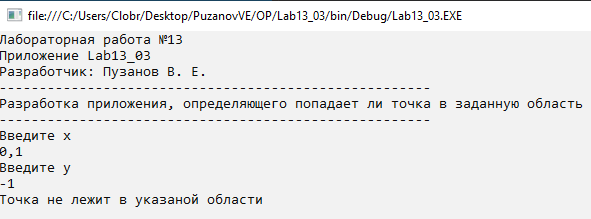


Рисунок 3.3 –Результат работы программы Lab13\_03

**Приложение Lab13\_04. Определение всех минимальных элементов**

***Задание*:** Разработать функцию, реализующую вычисление факториала целого неотрицательного числа n. Разработать приложение вызывающее функцию, вычисляющую факториал числа. Предусмотрите ситуацию, когда функции будут переданы недопустимые параметры: отрицательное значение аргумента. В этом случае функция Fact() должна вернуть значение 0. Реализуйте в основном коде программы ввод исходных данных, вычисление факториала путем вызова функции Fact(). Выполните анализ значения, возвращенного функцией: если оно равно нулю, программа должна вывести сообщение о недопустимом значении аргумента, в противном случае – полученное значение факториала.

***Константы:*** отсутствуют.

***Переменные:*** F – переменная хранящая в себе значение интеграла; n – вводимое число, интеграл которого необходимо найти.

***Исходные данные:*** n;

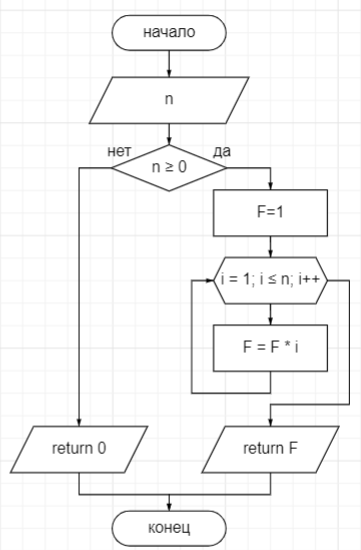
***Результат:***

Программа вычисляет интеграл вводимого числа.

***Блок-схема:***



***Блок-схема для функции Fact (n):***



***Код приложения:***

**Листинг 3 – Код консольного приложения Lab13\_04**

namespace PuzanovVE.OP.Lab13\_04

{

class Program

{

static int Fact(int n)

{

if (n >= 0)

{

int F = 1;

for (int i = 1; i <= n; i++)

{

F = F \* i;

}

return F;

}

else

{

return 0;

}

}

static void Main(string[] args)

{

//Стандартная шапка

Console.WriteLine("Лабораторная работа №13");

Console.WriteLine("Приложение Lab13\_04");

Console.WriteLine("Разработчик: Пузанов В. Е.");

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

//информация о задаче

Console.WriteLine("Разработка приложения, вычисляющего факториал числа");

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

//основной код программы

Console.WriteLine("Введите число, из которого нужно вычислить интеграл:");

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

if (Fact(n) > 0)

{

Console.WriteLine("Факториал числа " + n + " равен " + Fact(n));

}

else

{

Console.WriteLine("Некорректное число для вычисления факториала");

}

Console.ReadLine();

}

}

}

**Тестирование:**

Контрольный пример 1:

Исходные данные:

n = 5;

Результат:

Факториал числа 5 равен 120

Результат работы программы для указанных исходных данных приведён на рисунке 4.1

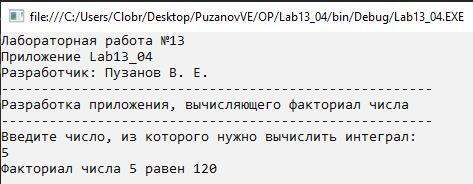


Рисунок 4.1 –Результат работы программы Lab13\_04

Контрольный пример 2:

Исходные данные:

n = -5;

Результат:

Некорректное число для вычисления факториала

Результат работы программы для указанных исходных данных приведён на рисунке 4.2

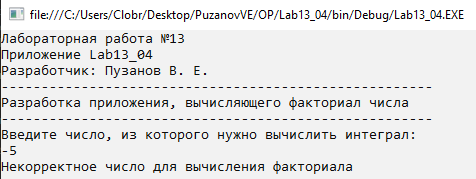


Рисунок 4.2 –Результат работы программы Lab13\_04

Контрольный пример 3:

Исходные данные:

n = 0;

Результат:

Факториал числа 0 равен 1

Результат работы программы для указанных исходных данных приведён на рисунке 4.3

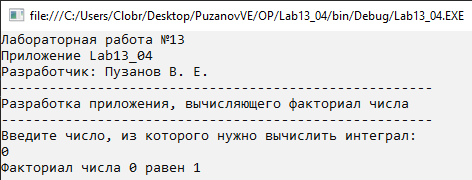


Рисунок 4.3 –Результат работы программы Lab13\_04

Выполнил студент Пузанов В. Е., ФИТУ 010304-КМСб-о22

Проверил ст. преподаватель каф. ПМ Черноиван Д.Н.