**НЕГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ ЧАСТНОЕ** **УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ** **«МОСКОВСКИЙ ФИНАНСОВО-ПРОМЫШЛЕННЫЙ УНИВЕРСИТЕТ** **“СИНЕРГИЯ”»**

|  |  |  |
| --- | --- | --- |
| **Факультет/Институт** |  |  |
|  |  | (наименование факультета/ Института) |
| **Направление/специальность** |  |  |
| **подготовки:** |  | (код и наименование направления /специальности подготовки) |
| **Форма обучения:** |  |  |
|  |  | (очная, очно-заочная, заочная) |
|  |  |  |

**Отчет по лабораторной работе №3**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **на тему** |  | Рекурсивные функции | | |
|  |  | (наименование темы) | | |
|  |  |  | | |
| **по дисциплине** | | |  | Разработка программных модулей |
|  | | |  | (наименование дисциплины) |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Обучающийся** |  | Рубцов Тимур Евгеньевич |  |  |
|  |  | (ФИО) |  | (подпись) |
| **Группа** |  | ДКИП-203 прог |  |  |
|  |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Преподаватель** |  | Сиберев И.В. |  |  |
|  |  | (ФИО) |  | (подпись) |

**Москва 2025 г.**

**Лабораторная работа №3. «**Рекурсивные функции**»**

Цель работы: изучить способы реализации алгоритмов с использованием рекурсии.

1.3. Индивидуальные задания.

Составить алгоритм в виде блок-схемы, написать и отладить поставленную задачу с использованием рекурсивной и обычной функций. Сравнить полученные результаты.

1. Для заданного целого десятичного числа N получить его представление в p-ичной системе счисления (p < 10).

Код:

using System;

class Program

{

static void Main()

{

Console.Write("Введите целое число N: ");

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

Console.Write("Введите систему счисления p (p < 10): ");

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

if (p >= 10 || N < 0)

{

Console.WriteLine("Ошибка: p должно быть < 10 и N >= 0");

return;

}

string iterativeResult = ConvertIterative(N, p);

string recursiveResult = ConvertRecursive(N, p);

Console.WriteLine($"\nРезультат (итеративно): {iterativeResult}");

Console.WriteLine($"Результат (рекурсивно): {recursiveResult}");

}

// Обычная (итеративная) функция

static string ConvertIterative(int n, int p)

{

if (n == 0) return "0";

string result = "";

while (n > 0)

{

result = (n % p) + result;

n /= p;

}

return result;

}

// Рекурсивная функция

static string ConvertRecursive(int n, int p)

{

if (n < p)

return n.ToString();

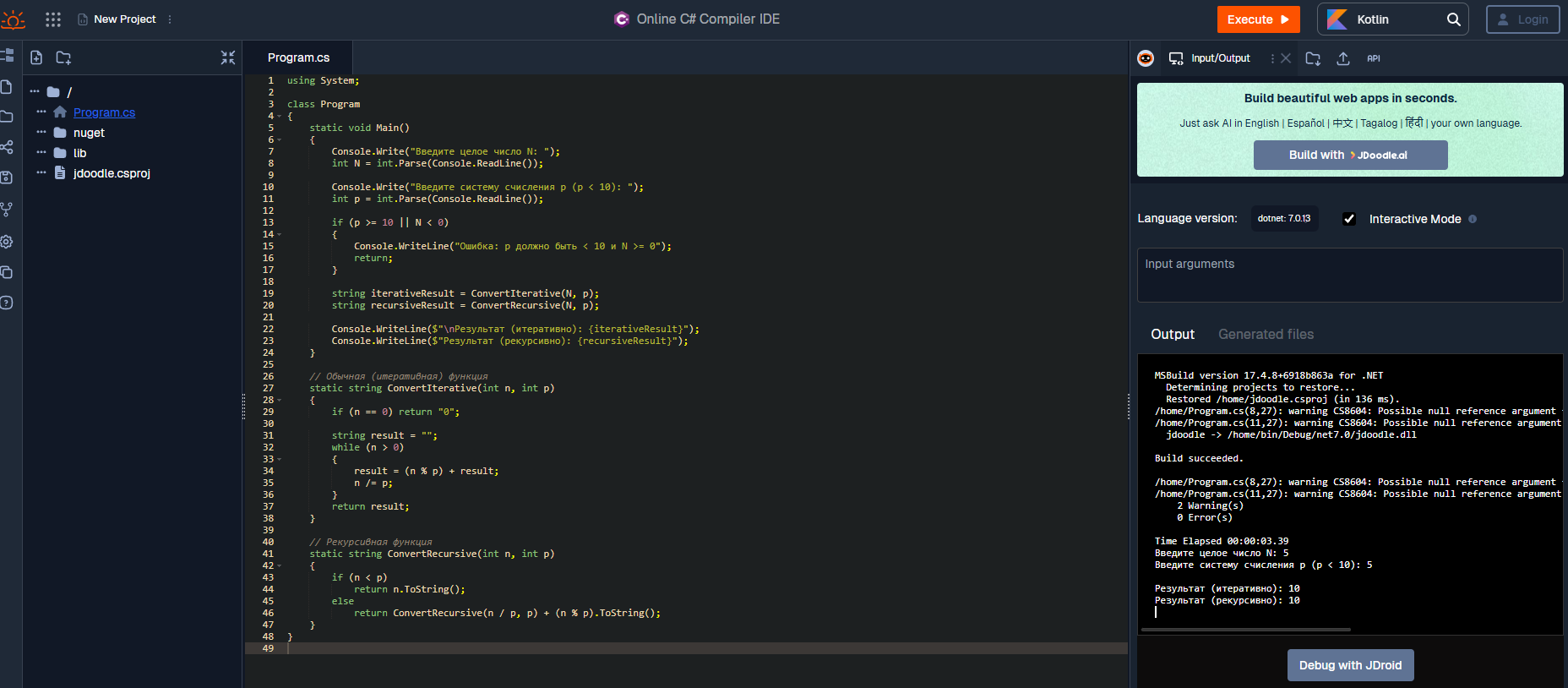
else

return ConvertRecursive(n / p, p) + (n % p).ToString();

}

}

Код на компиляторе:



1.4. Контрольные вопросы

1. Какая функция называется рекурсивной?

Рекурсивная функция — это функция, которая вызывает сама себя с другими аргументами. Она работает до тех пор, пока не будет достигнуто **условие завершения (база рекурсии)**.

2. Может ли в реализации рекурсивной функции существовать несколько операторов передачи управления return?

Да, может. Например, в базовом случае возвращается одно значение, а в общем — другая логика.