

ГУАП

КАФЕДРА № 43

ОТЧЕТ  
ЗАЩИЩЕН С ОЦЕНКОЙ

ПРЕПОДАВАТЕЛЬ:

ассистент

(должность, уч. степень, звание)

(подпись, дата)

Д. А. Кочин

(инициалы, фамилия)

ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ №1

«Рекурсия»

ПО КУРСУ: «Теория вычислительных процессов»

РАБОТУ ВЫПОЛНИЛ

СТУДЕНТ ГР. №

4031

подпись, дата

Х.В. Сидиропуло

инициалы, фамилия

Санкт-Петербург  
2023

## Цель работы

Преобразовать заданную арифметическую функцию в рекурсивную, используя операторы примитивной рекурсии.

Создать программу на языке высокого уровня содержащую две функции. Первая функция должна вычислять заданную арифметическую функцию итеративным способом. Вторая функция должна вычислять заданную арифметическую функцию рекурсивным способом.

## Основные сведения из теории

Рекурсия (от лат. *recursio* – бегу назад, возвращаюсь) – есть такой способ задания вычислимой функции, когда каждое ее значение задается через значение этой (или другой) вычислимой функции для меньших (или ранее определенных) значений аргументов, а функции задаваемые таким образом называются рекурсивными.

Оператор примитивной рекурсии позволяет конструировать  $n+1$  – местную функцию  $f$  по двум заданным вычислимым функциям:

$$\begin{aligned} g^n(x \dots x_n) & - n \text{ местная функция} \\ h^{n+2}(x \dots x_n) & - n + 2 - \text{местная функция} \\ \text{где для всех } x_1 \dots x_n, y & \in N \end{aligned}$$

## Преобразование арифметической функции в рекурсивную с использованием операторов примитивной рекурсии

Вариант:  $x/4$

$$\begin{aligned} f(x, y) &= x/y // \text{примитивно рекурсивная функция} \\ f(x, 0) &= x * 0 = 0 // \text{базовый случай} \\ f(x, y) &= g(f(x-1, y), 1/y) = g(f(x-1, y) + 1/y) \\ &= g(f(x-1, 4) + 1/4) // \text{рекурсивный случай} \end{aligned}$$

## Листинг программы на языке высокого уровня с комментариями

```
namespace CodeConversion
{
    class Program
    {
        static decimal g(decimal x)
        {
            return x;
        }

        static decimal iter(int x, decimal y)
        {
            return x / y;
        }
        static decimal f(int x, decimal y)
        {
            if (x <= 0)
            {
```

```

        return x / y;
    }

    decimal result = f(x - 1, y);
    return g(result + 1 / y);
}
static void Main(string[] args)
{
    decimal x;
    Console.WriteLine(" Ввести x: ");
    x = decimal.Parse(Console.ReadLine());
    decimal y = 4.0m;
    decimal result = f((int)x, y);
    decimal iterResult = iter((int)x, y);
    Console.WriteLine("Рекурсивный результат f(" + x + ", " + y + "): " + result);
    Console.WriteLine("Итеративный результат f(" + x + ", " + y + "): " + iterResult);
}
}

```

### Пример результата выполнения

Консоль отладки Microsoft Visual Studio	
Ввести x:	Ввести x:
16	-24
Рекурсивный результат f(16, 4,0): 4,00	Рекурсивный результат f(-24, 4,0): -6
Итеративный результат f(16, 4,0): 4	Итеративный результат f(-24, 4,0): -6

## **Вывод**

В ходе лабораторной работы ознакомились с принципом решения арифметической функции рекурсивным методом.