Министерство науки и высшего образования Российской Федерации

**ФГБОУ ВО «Кубанский государственный технологический университет»**

**(ФГБОУ ВО «КубГТУ»)**

Институт компьютерных систем и информационной безопасности

Кафедра информационных систем и программирования

Направление 10.05.01 Компьютерная безопасность

**ОТЧЕТ по практическому занятию №7**

по дисциплине: «Языки программирования» на тему:

«Рекурсия.»

Выполнил студент: Рудометкин Станислав Владимирович, 1 курс, группа 22-К-КБ1

Допущена к защите\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Руководитель работы,

профессор Г.С. Петриченко

Краснодар

2023

**ПРАКТИЧЕСКАЯ РАБОТА №7**

**Тема**: Рекурсия

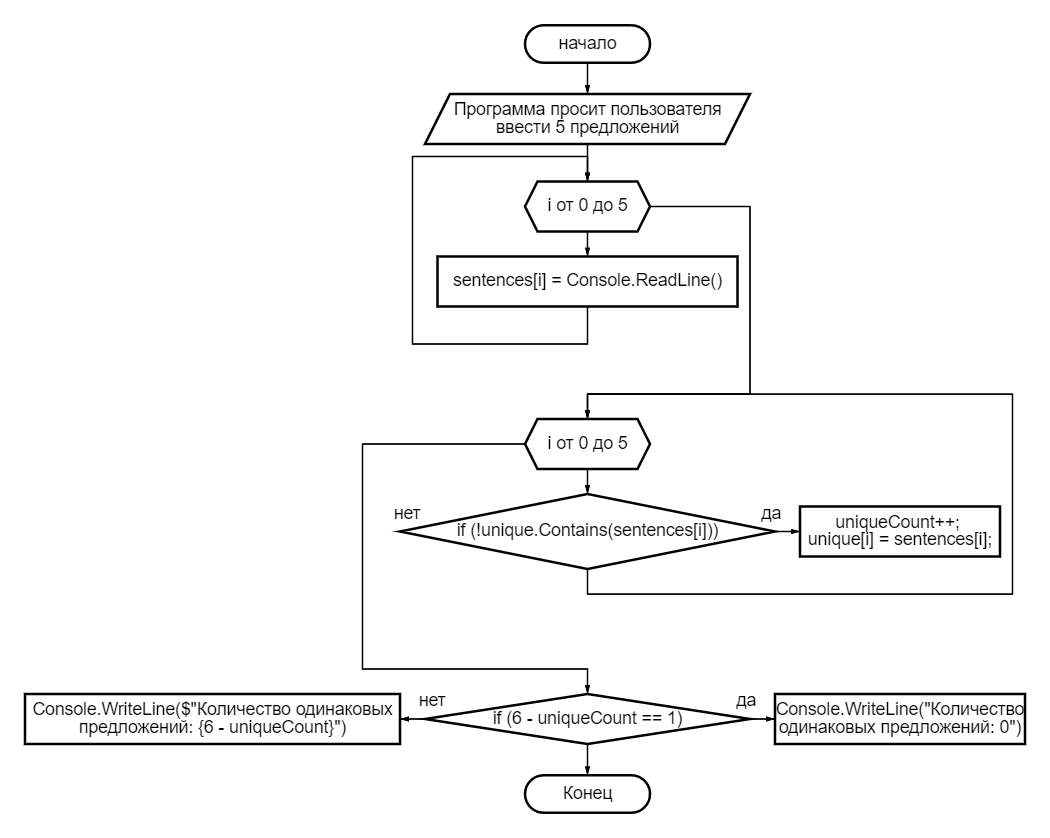
**Цель**: изучить принципы работы рекурсивных алгоритмов и научиться

применить полученные занятия на практике.

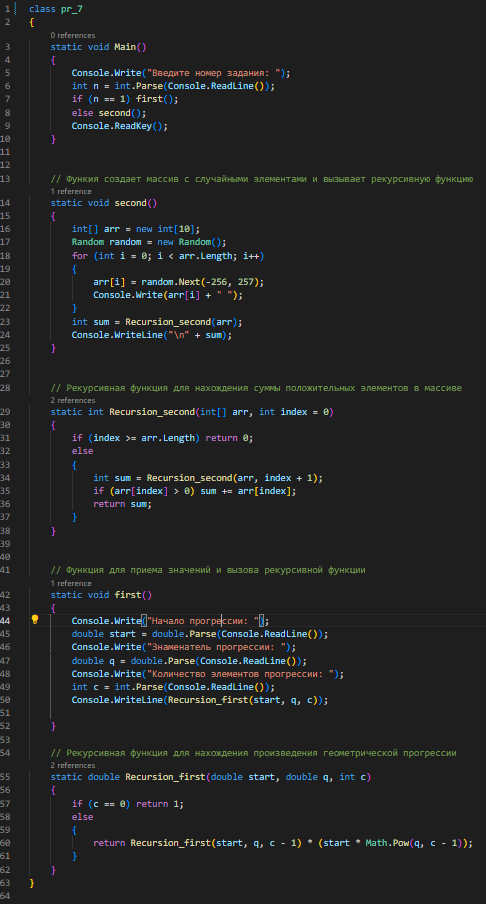
**Задание (вариант 14):**

**** ****

**Алгоритм схемы:**



**Код программы Скриншот**



**Исходный код программы**

class pr\_7

{

static void Main()

{

Console.Write("Введите номер задания: ");

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

if (n == 1) first();

else second();

Console.ReadKey();

}

// Функия создает массив с случайными элементами и вызывает рекурсивную функцию

static void second()

{

int[] arr = new int[10];

Random random = new Random();

for (int i = 0; i < arr.Length; i++)

{

arr[i] = random.Next(-256, 257);

Console.Write(arr[i] + " ");

}

int sum = Recursion\_second(arr);

Console.WriteLine("\n" + sum);

}

// Рекурсивная функция для нахождения суммы положительных элементов в массиве

static int Recursion\_second(int[] arr, int index = 0)

{

if (index >= arr.Length) return 0;

else

{

int sum = Recursion\_second(arr, index + 1);

if (arr[index] > 0) sum += arr[index];

return sum;

}

}

// Функция для приема значений и вызова рекурсивной функции

static void first()

{

Console.Write("Начало прогрессии: ");

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

Console.Write("Знаменатель прогрессии: ");

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

Console.Write("Количество элементов прогрессии: ");

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

Console.WriteLine(Recursion\_first(start, q, c));

}

// Рекурсивная функция для нахождения произведения геометрической прогрессии

static double Recursion\_first(double start, double q, int c)

{

if (c == 0) return 1;

else

{

return Recursion\_first(start, q, c - 1) \* (start \* Math.Pow(q, c - 1));

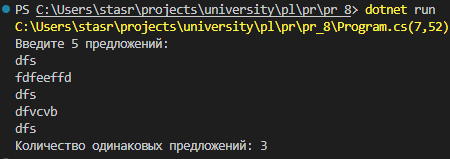
}

}

}

**Вывод результатов работы программы**

**Скриншот**



**Пояснение работы программы**

Данный код на C# содержит две рекурсивные функции и одну функцию, вызывающую одну из двух рекурсивных функций в зависимости от введенного пользователем значения.

Функция "Recursion\_first" находит произведение геометрической прогрессии. Она принимает начальное значение "start", знаменатель "q" и количество элементов "c" в прогрессии. Если количество элементов равно 0, функция возвращает 1. В противном случае функция вызывает саму себя с уменьшенным на 1 значением "c" и возвращает произведение "start" и "q" в степени "c-1", умноженное на результат вызова самой себя.

Функция "Recursion\_second" находит сумму положительных элементов в массиве. Она принимает массив "arr" и начальный индекс "index" (по умолчанию равный 0). Если индекс превышает размер массива, функция возвращает 0. В противном случае функция вызывает саму себя с индексом, увеличенным на 1, и сохраняет результат в переменную "sum". Если значение элемента массива с индексом "index" положительно, оно добавляется к переменной "sum". В конце функция возвращает переменную "sum".

Функция "second" создает массив "arr" длиной 10 со случайными целочисленными элементами в диапазоне от -256 до 256. Затем функция вызывает функцию "Recursion\_second", передавая ей массив "arr" в качестве аргумента, и сохраняет результат в переменной "sum". В конце функция выводит на экран элементы массива и значение переменной "sum".

Функция "first" запрашивает у пользователя начальное значение прогрессии "start", знаменатель "q" и количество элементов в прогрессии "c". Затем функция вызывает функцию "Recursion\_first", передавая ей "start", "q" и "c" в качестве аргументов, и выводит результат на экран.

Функция "Main" запрашивает у пользователя номер задания и вызывает функцию "first" или "second" в зависимости от введенного значения. В конце функция ожидает ввода пользователем любой клавиши для завершения программы.