# 63-64 ОАП Пр14 Применение рекурсивных функций.

# Теоретические сведения

**Рекурсивные определения функций.** В С++ допускается рекурсивное определение функций. Проиллюстрируем определение рекурсивной функции на классическом примере вычисления факториала целого положительного числа.

**long** Factor(**int** n)  
{  
 **if**(n<0) **return** 0;  
 **if**(n==0) **return** 1;  
 **return** n\*Factor(n-1);  
}

В случае если при вызове функции будет задан отрицательный аргумент, она вернет нулевое значение — признак неверного обращения. Если n = 0, то функция вернет 1. Иначе, функция рекурсивно вызывает саму себя с параметром, меньшим на 1.

## Контрольные вопросы

1. На чем основан рекурсивный метод программирования?
2. В чем разница между «циклическим» и «рекурсивным» способами определения? Какой элемент является обязательным в рекурсивном определении?
3. К каким последствиям приводит «рекурсивное зацикливание»?
4. Какое условие должно обязательно присутствовать в любой рекурсивной функции?
5. Что такое явная и косвенная рекурсии?
6. Дайте рекурсивное определение целой степени числа N.

## Задания к практической работе № 10

1. Ввести последовательность чисел (окончание ввода – 0) и вывести их в обратной последовательности. Входные данные взять из текстового файла.
2. Используя cout <<x лишь при x=0..9, написать рекурсивную программу печати десятичной записи целого положительного числа n.
3. Напишите рекурсивную функцию, которая возвращает среднее из n элементов массива чисел.
4. Найти первые N чисел Фибоначчи двумя способами: с помощью рекурсии и с помощью итерации. Сравнить эффективность алгоритмов.
5. Написать функцию сложения двух чисел, используя только прибавление единицы.
6. Написать функцию умножения двух чисел, используя только операцию сложения.
7. Вычислить сумму элементов одномерного массива.
8. Найти НОД (наибольший общий делитель) двух натуральных чисел.
9. Вычислить несколько значений функции Аккермана для неотрицательных чисел m и n:



1. Напишите рекурсивную функцию, которая вычисляет длину строки.
2. Написать функцию C(m,n) вычисления биномиальных коэффициентов  по следующей формуле: 
3. Проверить, является ли фрагмент строки с *i*-го по *j*-й символ палиндромом.
4. Вычислить произведение элементов одномерного массива.
5. Написать функцию сортировки массива методом простого выбора.
6. Подсчитать количество цифр в заданном числе.
7. Написать функцию, проверяющую правильность имени в языке Pascal.
8. Вычислить определитель матрицы, пользуясь формулой разложения по первой строке:  
     
   где матрица Bk получается из A вычеркиванием первой строки и k-го столбца.



1. Написать функцию определения, является ли заданное натуральное число простым.
2. Реализовать рекурсивный алгоритм построения цепочки из имеющегося набора костей домино.
3. Расстояния между городами заданы матрицей (Если между городами i,j есть прямой путь с расстоянием N, то элементы матрицы A(i,j) и A(j,i) содержат значение N, иначе 0). Написать программу поиска минимального пути обхода всех городов без посещения дважды одного и того же города (задача коммивояжера).
4. Задан массив целых. Построить из них любую последовательность таким образом, чтобы последняя цифра предыдущего числа совпадала с первой цифрой следующего.
5. Для данного n напечатайте коэффициенты разложения полинома (1 + x)n.
6. Вычислить, используя рекурсию, выражение



1. Написать функцию печати всех перестановок из n символов.
2. Написать функцию перевода числа из десятичной системы счисления в двоичную.