**Практическая работа №1.**

**1) Рекурсия в широком смысле** – это способ определения функций, являющийся объектом изучения в теории алгоритмов и других разделах математической логики. Этот способ давно применяется в арифметике для определения числовых последовательностей (прогрессии, чисел Фибоначчи и пр.). Существенную роль играет Р. в вычислительной математике (рекуррентные методы). Наконец, в теории множеств часто используется трансфинитная Р.

**2)Примеры рекурсии из жизни:**

1. Матрешки: каждая матрешка содержит внутри себя еще одну матрешку, которая в свою очередь также содержит еще одну матрешку, и так далее.

2. Фракталы: многие геометрические фигуры, такие как ковер Серпинского или треугольник Серпинского, являются рекурсивными. Каждая часть фигуры содержит в себе меньшую копию этой же фигуры.

3. Деревья: деревья имеют иерархическую структуру, где каждый узел может содержать множество других узлов-потомков, которые в свою очередь также могут содержать другие узлы-потомки.

4. Функции: в программировании функции могут вызывать сами себя, чтобы выполнить определенную задачу. Это называется рекурсивной функцией.

5. Генеалогические деревья: при составлении генеалогических деревьев каждый человек имеет родителей, которые в свою очередь также имеют родителей, и так далее. Это создает рекурсивную структуру.

**3) Рекурсивный алгоритм** - это алгоритм, который может вызывать сам себя во время своего выполнения. Вместо того, чтобы решать задачу напрямую, рекурсивный алгоритм разбивает ее на более простые подзадачи, которые решаются с использованием того же алгоритма. Процесс повторяется до тех пор, пока подзадачи не станут достаточно простыми для непосредственного решения. Рекурсивные алгоритмы часто используются для решения задач, которые могут быть естественно разделены на подзадачи, такие как обход дерева, вычисление факториала или сортировка.

**4) Рекурсивная триада** - это группа из трех элементов, каждый из которых содержит ссылку на другой элемент в этой же группе. Таким образом, каждый элемент является частью циклической структуры, которая может быть определена рекурсивно. Примером рекурсивной триады может быть группа из трех людей, где каждый человек является родителем для одного из оставшихся двух людей.

**1.параметризация** – выделяют характеристики, которые используются для описания условия задачи, а затем в решении;

**2.база рекурсии** – определяют простой случай, при котором решение понятно, то есть не нуждается в обращение функции к себе;

**3.декомпозиция** – выражают общий случай через более лёгкие подзадачи с измененными значениями.

**5)** **Полное дерево** - это дерево, в котором каждый узел имеет либо ноль, либо два дочерних узла. В полном дереве все уровни, кроме, возможно, последнего, полностью заполнены.

**Глубина дерева** - это количество уровней в дереве. Уровень 0 состоит только из корневого узла, уровень 1 состоит из дочерних узлов корня, и так далее. Глубина дерева определяет количество шагов, необходимых для достижения определенного узла из корня.

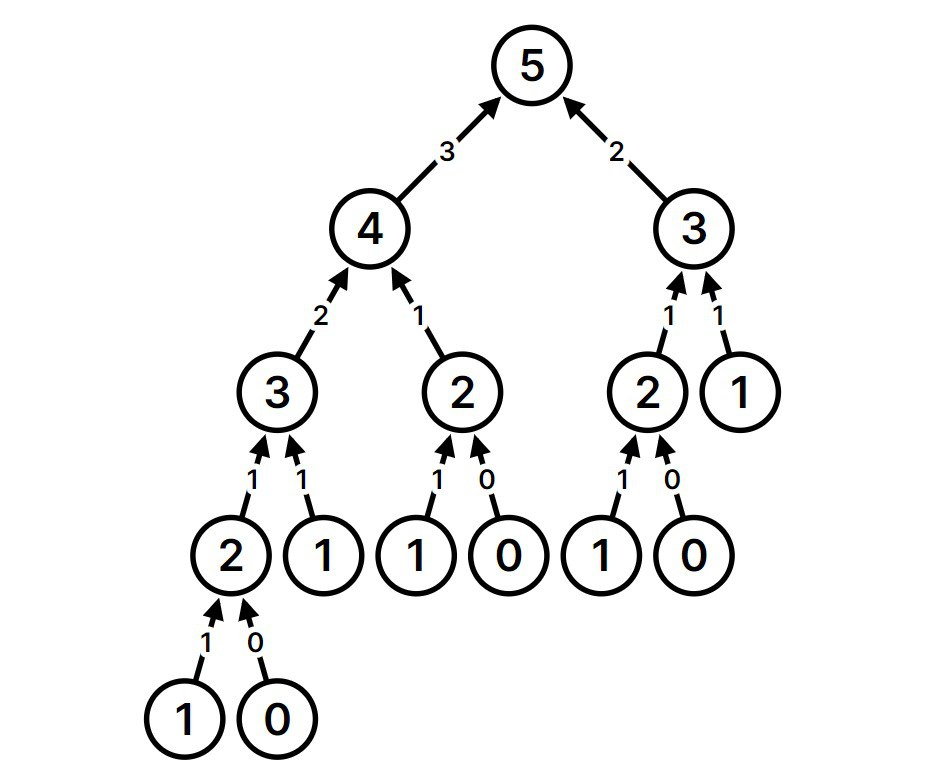
**Объем рекурсии** - это количество вызовов функции, которые могут быть активными одновременно во время выполнения рекурсивного алгоритма. Каждый вызов функции добавляет новый уровень рекурсии. Если рекурсивный алгоритм не управляется должным образом, объем рекурсии может стать очень большим и привести к переполнению стека или другим проблемам производительности.

**6)** Область памяти, выделяемая для хранения всех промежуточных значений локальных переменных, называется **стеком (stack).**

**Стек** (англ. stack — стопка; читается стэк) — абстрактный тип данных, представляющий собой список элементов, организованных по принципу LIFO (англ. last in — first out, «последним пришёл — первым вышел»).

**7)** **Рекурсивная функция** - это функция, которая вызывает саму себя. Вот пример рекурсивной функции на языке Python, вычисляющей факториал числа:

//Python  
def factorial(n): //параметризация  
 if n == 1: //база  
 return 1  
 else:  
 return n \* factorial(n-1) //декомпозиция



Глубина 5, объем 14.

**Рекурсивная триада** - это концепция, связанная с рекурсивными функциями. Она состоит из трех частей, которые характеризуют рекурсивный процесс:

1. **Базовый случай**: если n равно 1, то функция возвращает

2. **Прогрессирующий случай:** в противном случае функция возвращает n, умноженное на вызов функции factorial с аргументом n-1.

3. **Рекурсивный случай**: вызов функции factorial внутри самой себя.

Использование этих трех составляющих позволяет понять и описать логику и поведение рекурсивной процедуры.

8)МЕМ ПЕРВЫЙ: МЕМ ПРО РЕКУРСИЮ



МЕМ ВТОРОЙ: МЕМ ПРО РЕКУРСИЮ С КОТОМ  


Объяснение: Действия имеют замкнутую цепочку последовательности. И когда 1 круг подходит к концу то начинается второй.