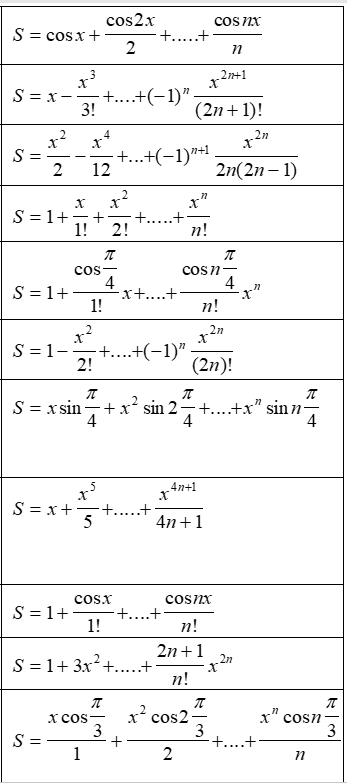
|  |  |
| --- | --- |
| Министерство науки и высшего образования Российской Федерации  Федеральное государственное автономное образовательное учреждение высшего образования  **«Пермский национальный исследовательский политехнический университет»** | |
| Электротехнический факультет  Кафедра «Информационные технологии и автоматизированные системы»  направление подготовки: 09.03.04 – «Программная инженерия» | |
| **Лабораторная работа №2**  **Рекурсивные функции**  **Вариант 7** | |
|  | Выполнила студентка гр. РИС-24-1б  Сабирзянова Анастасия Маратовна |
|  | Проверил:  Доц. каф. ИТАС  Ольга Андреевна Полякова  \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_  *(оценка)*  *(подпись)*  *(дата)* |
| г. Пермь, 2024 | |

**Постановка задачи**

Необходимо найти сумму последовательности с использованием рекурсивных функций на языке С++:

**Основная часть**

Рекурсия— это способ определения функции, при котором функция вызывает саму себя в процессе выполнения.

Основные идеи рекурсии:

1. Базовый случай (Base Case). Это условие, при котором рекурсия останавливается. Без базового случая рекурсия будет продолжаться бесконечно, что приведет к переполнению стека (stack overflow).
2. Рекурсивный шаг (Recursive Step). Это часть функции, где она вызывает саму себя. Рекурсивный шаг обычно разбивает исходную задачу на подзадачи, которые могут быть решены с помощью того же алгоритма, но на меньших объемах данных.

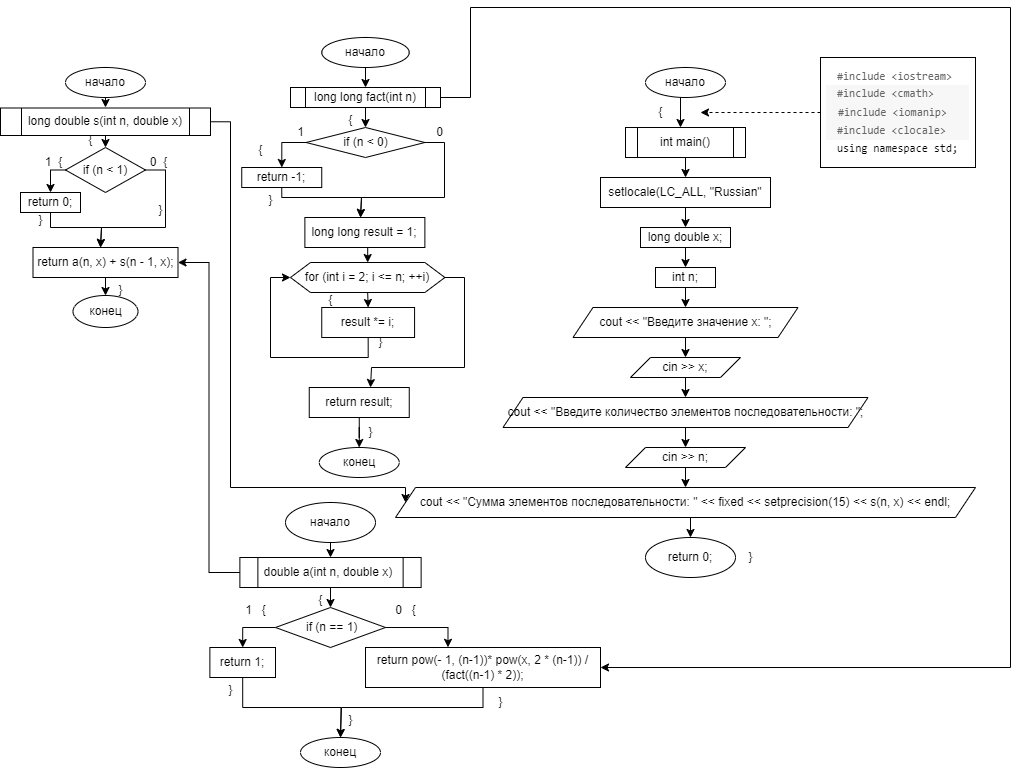
Данную задачу можно решить при помощи этого способа.

Анализ задачи:

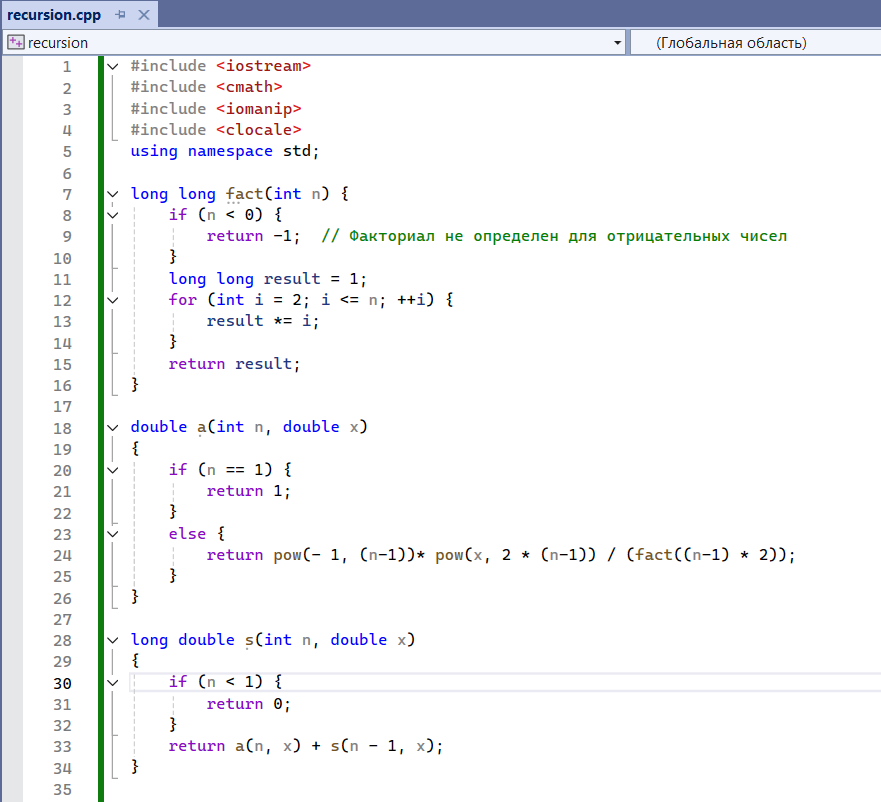
1. Сначала вводим функцию для подсчета факториала.
2. С помощью рекурсии считаем, чему равен текущий элемент последовательности.
3. Запоминаем его и прибавляем к нему предыдущий элемент, для этого вызываем ту же функцию с n, меньшим на 1.
4. Повторяем до достижения n = 0.
5. Учитываем, что последовательность начинается с n=0, а 0 элементов не могут выдать нам сумму последовательности.

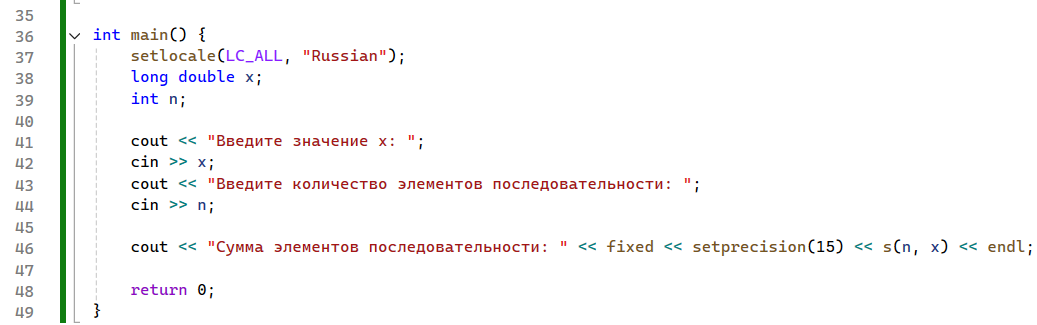
**Решение на языке С++**

**Блок-схема:**

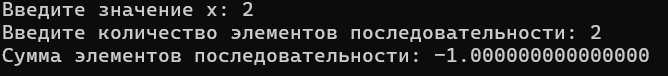
****

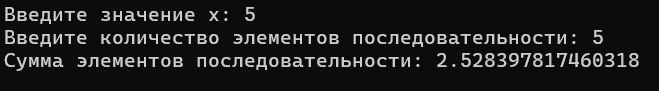
**Код:**

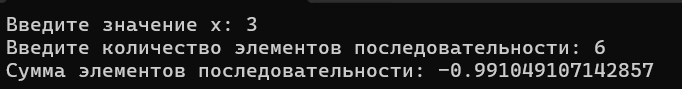
****

****

**Вывод программы:**

****

****

****