## 1. Ход работы

## 1.1. Код приложения

```
from decimal import Decimal, getcontext

# Установка точности вычислений
getcontext().prec = 50

def add_long_numbers(num1, num2):
    # Преобразование чисел в длинный формат Decimal
    decimal_num1 = Decimal(num1)
    decimal_num2 = Decimal(num2)

# Сложение длинных чисел
    result = decimal_num1 + decimal_num2

# Вывод результата
    print(f"Результат сложения: {result}")

# Пример использования функции
num1 = "123456789012345678901234567890"
num2 = "9876543210987654321098765432109876543210"
add_long_numbers(num1, num2)
```

## 1.2. Теория

Умножение Карацубы Метод умножения Карацубы относится к парадигме «разделяй и властвуй».

Данный алгоритм представляет собой простую реализацию идеи разделения входных данных, которая стала базисной для нижеперечисленных алгоритмов. Идея заключается в разделении одной операции умножения над n- значными числами на три операции умножения над числами длины n/2 плюс O(n). Для перемножения двух чисел, превышающих длину машинного слова, алгоритм Карацубы вызывается рекурсивно до тех пор, пока эти числа не станут достаточно маленькими, чтобы их можно было перемножить непосредственно.

## 2. Вставка изображения результата отладки

Отладка программы представлена на рис.1.

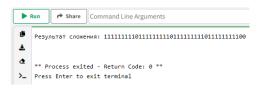


Рис. 1. Отладка программы