**Идея приложения.**

Калькулятор, который на вход принимает целое выражение в виде строки, например, 5 + 3 - 2, а по нажатию на кнопку Enter - вычисляет ответ. Кроме того, калькулятор должен распознавать в входном выражении какие-либо заранее заданные константы, написанные буквами, а не числами, например, "pi" - 3.14. По итогу, для выражения "5 + pi + e" - ответ должен быть, при условии, что "pi" - 3.14, "e" - 2.71, - "10.85".

Упрощения:

1. Поддерживаемые математические операции: "+" и "-";

2. Отступ между оператором и операндом - 1 пробел;

3. Вещественные числа пишутся через точку;

4. Текстом можно писать только заранее заданные константы;

5. Скобки не поддерживаются;

6. Любое отклонение от описанных выше пунктов - ошибка "Неверное выражение".

**Выбор паттерна.**

На первый взгляд может показаться, что паттерна "Компоновщик" должно быть достаточно для решения данной задачи, так как, по сути, математическое выражение, это есть композиция более мелких выражений и т.д. Подобная структура будет обладает хорошей гибкостью: если мы захотим добавить новый вид выражений, нам достаточно унаследовать новый класс от базового интерфейса выражения. Однако недостатком как этого, так и других структурных паттернов заключается в том, что они не описывают способ, среду взаимодействия элементов структуры.

А конкретно, нам нужно, чтобы наша программа понимала, что слово “pi”, это есть 3.14 и никак иначе. То есть, чтобы элементы нашей структуры знали о существующем контексте и зависели от него при выполнении операций. И единственным паттерном, который удовлетворял бы нашим ожиданиям является паттерн “Интерпретатор”.

Что не удивительно, так как его определение, говорит, что он был специально создан для решения подобных проблем, а именно для организации процесса синтаксического разбора и интерпретации лингвистических конструкций. Паттерн “Интерпретатор” определяет грамматику языка, представляет грамматические правила в виде языковых предложений и интерпретирует их для решения задачи. Для представления каждого грамматического правила паттерн Интерпретатор использует отдельный класс.