

Слущкий Никита. 053501. Лабораторная Работа №12. Реализация калькулятора

Реализован класс Calculator. Он позволяет производить вычисления значений выражений с учётом приоритетов, скобок. На вход класс получает строку, состоящую из чисел; скобок '(', ')'; знаков '+', '-', '/', '*', '^'; пробелов (для удобства читаемости при вводе). Таким образом из умений: сложение, вычитание, деление, умножение и возведение в степень.

В классе присутствуют следующие поля:

- PRIORITIES — статический (то есть общий для всех объектов класса) ассоциативный массив с указанием приоритетов операций
- DIGITS — статическая «коллекция» существующих в природе цифр
- OPERATORS — статическая «коллекция» поддерживаемых знаков
- BRACKETS — то же самое для скобок (на самом деле только одного типа)
- originalString — «сырая» непросчитанная строка, получаемая на входе при инициализации калькулятора
- answer — будущая переменная для хранения ответа. Не считать же ответ каждый раз, поэтому после первого просчёта он просто запоминается
- isAlreadyComputed — вот как раз переменная, отвечающая за контроль того, было ли уже обработано и посчитано выражение
- numbers — стек для чисел, нужный в процессе вычисления значения выражения
- signs — такой же стек, но для операторов

В классе присутствуют следующие методы

- Trim () — вспомогательный метод для игнорирования пробелов во время анализа строки
- Pow () — метод возведения в степень (чтобы не подключать cmath и обойтись своими силами)
- SimpleCalculation () — метод для произведения простых вычислений для двух чисел (+, -, /, *, ^)
- GetNumberFromString () — метод выделения числа из строки
- ProcessOperator () — метод для обработки знаков -, /, *, +, ^. Контролирует приоритеты, вычисляет по мере поступления более низкоприоритетных знаков
- 2 конструктора и деструктор (не методы, но упомянуть надо)
- CheckingCorrectnessOfBracketSequence () — это «пасхалка». В любом калькуляторе должна быть проверка на правильную скобочную последовательность. Это применение знаний из одной предыдущей ЛР
- Compute () — главный метод вычисления
- Answer () — возвращает ответ на пример. Если ещё не посчитано, то он запустит вычисление и только потом вернёт ответ
- Clear () — очищает все рабочие стеки, приводит всё в «первозданный» вид
- ChangeExpression () — возможность смены примера в рамках одного и того же экземпляра класса Calculator

Перегружены операторы:

- Потокового вывода <<cout — вывод в виде «выражение = ответ»
- Арифметических действий — чтобы можно было сразу вывести результат одного из простых действий между двумя объектами Калькулятора. То есть есть два объекта (две строки с вычислениями) — можно найти результат базового действия (+, -, *, /, ^) между этими двумя объектами