

# Лабораторная работа № 6 по курсу дискретного анализа: калькулятор

Выполнил студент группы 08-308 МАИ *Шарапов Леонид*.

## Условие

Составить программу, использующую программную библиотеку, реализующую простейшие арифметические действия и проверку условий над целыми неотрицательными числами.

## Метод решения

1. Реализован класс длинного числа с перегрузками требуемых арифметических операций и условий
2. Перегружены операции поточного ввода и вывода
3. Основание системы счисления равняется 10000
4. Считываются длинные числа, а затем требуемая операция

## Описание программы

Весь код находится в `main.cpp`

Используемые типы данных

1. `std::vector<int>` num - хранение длинного числа
2. `const int` - основание системы счисления и количество нулей в ней
3. `int`
4. `std::string` - введенное число
5. `char` - операция
6. `longNumber` - длинное число

Используемые функции

1. `main` - считывание чисел и операции, а затем выбор действий
2. `longNumber::longNumber(int n)` - конструктор для считывания длинного числа через `int`

3. `std::istream& operator>(std::istream& in, longNumber& obj)` - перегрузка поточного ввода
4. `std::ostream& operator<(std::ostream& out, longNumber obj)` - перегрузка поточного вывода
5. `longNumber longNumber::operator+(const longNumber& right) const` - перегрузка операции сложения длинных чисел
6. `void longNumber::delLeadingZeros()` - удаление незначащих нулей из длинного числа после выполнения операции
7. `longNumber longNumber::operator-(const longNumber& right) const` - перегрузка операции вычитания длинных чисел
8. `longNumber longNumber::operator*(const longNumber& right) const` - перегрузка операции умножения длинных чисел
9. `longNumber longNumber::operator/(const longNumber& right) const` - перегрузка операции деления длинных чисел
10. `longNumber longNumber::operator^(const longNumber& right) const` - перегрузка операции возведения в степень длинного числа
11. `longNumber binPow(const longNumber& right) const` - рекурсивное быстрое возведение в степень
12. `bool longNumber::operator==(const longNumber& right) const` - перегрузка оператора равенства длинных чисел
13. `int longNumber::cmp(longNumber& obj) const` - функция сравнения длинных чисел, возвращающая 0, если они равны, иначе -1 или 1, если первое больше второго и наоборот

## Дневник отладки

1. `time limit 10pow.t` - для 0 в степени  $x$  была долгая работа при больших  $x > 0$ . Введено условие, чтобы было быстрее
2. `time limit 10pow.t` - переделано бинарное возведение в степень из итеративной формы в рекурсивную
3. `wrong answer 3minus.t` - при вычитании из 10000 единицы  $l - r + k$  становилось меньше нуля. Исправлено
4. `wrong answer 6pow.t` - для 0 в степени 0 должна выводиться ошибка

## Выводы

Длинная арифметика используется для вычислений на ПК, потому что типы переменных в них ограничены числом, то есть для избежания переполнения. Фундаментальные математические константы, например,  $\pi$  могут быть вычислены с ее помощью. Основными проблемами были долгий поиск теории, работа с ней и техническим заданием.