

Лабораторная работа № 1 по курсу ООП: работа с классами

Выполнил студент группы 08-208 МАИ *Попов Данила*.

Условие

Задание №1: написать класс, который реализует комплексное число в алгебраической форме со следующими функциями:

1. Сложение
2. Вычитание
3. Умножение
4. Деление
5. Сравнение
6. Сопряжённое число

Метод решения

Общее описание алгоритма решения задачи, архитектуры программы и т. п. Полностью расписывать алгоритмы необязательно, но в общих чертах описать нужно. Приветствуются ссылки на внешние источники, использованные при подготовке (книги, интернет-ресурсы).

Описание программы

Код программы состоит из 3-х файлов:

1. `app/main.cpp`: файл, содержащий точку входа приложения
2. `include/complex.hpp`: файл, содержащий объявление и реализацию inline-функций
3. `src/lib/complex.cpp`: реализация не-inline методов класса `Complex.h`

Дневник отладки

Была одна очень забавная проблема в `test.py`. После считывания тестового запроса и отправки его в программу, весь тест зависал. Как оказалось, проблема была в отсутствующем 'n' в конце запроса, который породил немало головной боли :)

Недочёты

Метод `Str()` работает не с внешним буффером, а создаёт каждый раз минимум два:

1. Для `std::stringstream` объекта, который предоставляет удобный интерфейс приведения стандартных типов к строке.
2. Для `std::string` объекта, который является возвращаемым значением.

Данный метод может использовать достаточно много процессорного времени при приведении объектов типа `Complex` к строке.

Выводы

В целом, повторил синтаксис написания классов в `C++`, изучил базовое межпроцессорное взаимодействие через пайпы и отловил один коварный баг, который не так просто обнаружить. Так же вспомнил, что именно делает модификатор `inline`, который используется по умолчанию при объявлении с реализацией методов в классе.