**Санкт-Петербургский государственный политехнический университет**

Кафедра “Прикладная математика”

Отчет по лабораторной работе №2 (вариант №4)

“Алгоритмы и структуры данных”

Студент группы № 5030102/20001

Соколов Артем Николаевич

Выполнено: 30.10.2023

**Постановка задачи**

Написать библиотеку работы с полиномами произвольной степени, позволяющую осуществлять их сложение, вычитание, умножение. Исходные данные зачитывать из файла.

**Описание алгоритма и структур данных**

Структуры:

1. “class Polynom” (в библиотеке "polyLIB.h") - класс Полинома, который требуется реализовать.
   1. Для класса Polynom определены 2 конструктора:
      1. explicit Polynom(int size) noexcept - конструктор, принимающий в себя только длину будущего полинома. При инициализации массив с коэффициентами заполняется нулями.
      2. explicit Polynom(const std::string& filename) - конструктор, считывающий полином из файла.
   2. Деструктор очищает массив коэффициентов.
   3. Для класса переопределен оператор [], который возвращает коэффициент перед (x^index). Если index был передан неверно, то выбрасывает исключение.
   4. Метод getLength() возвращает длину полинома.
   5. Метод get() возвращает коэффициент перед (x^index). Если index был передан неверно, то вернет 0.
   6. Метод print() выводит полином в консоль.
   7. Также для объектов класса были переопределены арифметические операторы “плюс”, “минус” и “умножить”. Результатом операции будет полином типа максимального веса из типов используемых полиномов.
2. “typedef struct list {...} List” - локальный (для метода explicit Polynom(const std::string& filename)) тип односвязного списка. Нужен для чтения полинома из файла, т.к. в классе коэффициенты хранятся в массиве от 0 ого к Тому, а в файле напротив, от N ого к 0 ому.

**Текст программы:**

[algos/2lab-4var at main · azya0/algos (github.com)](https://github.com/azya0/algos/tree/main/2lab-4var)

и/или

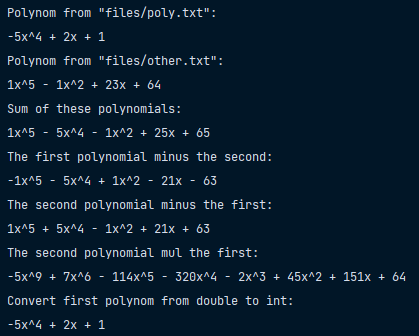
https://github.com/azya0/algos/tree/main/2lab-4var

**файл main.cpp**

**Описание тестирования**

Для тестирования данной программы производился ее неоднократный запуск с различным содержимым текстового файла.

Пример тестирования:



В ходе тестирования программы не было выявлено ошибок.