МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение

высшего образования

**«Нижегородский государственный университет им. Н.И. Лобачевского»**

**Национальный исследовательский университет**

**Институт информационных технологий, математики и механики**

**Кафедра математического обеспечения и суперкомпьютерных технологий**

**Отчет по учебной практике**

**«Система для арифметических действий над многочленами нескольких переменных»**

**Выполнил:** студент группы 381706-1

Полетуева Анастасия Николаевна

\_\_\_\_\_\_\_\_\_\_\_ Подпись

**Научный руководитель:**

ассистент каф. МОСТ ИИТММ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Лебедев И.Г

Нижний Новгород

2018.

Содержание

[1. Введение 3](#_Toc536801925)

[2. Постановка задачи 4](#_Toc536801926)

[3. Руководство пользователя 5](#_Toc536801927)

[4. Руководство программиста 6](#_Toc536801928)

[4.1. Описание структуры программы 6](#_Toc536801929)

[4.2. Описание структур данных 6](#_Toc536801930)

[4.3. Описание алгоритмов 8](#_Toc536801931)

[5. Заключение 10](#_Toc536801932)

# Введение

**Основная цель данной работы** – разработка системы для выполнения арифметических операций над полиномами.

**Mоном** - произведение, состоящее из постоянного числового множителя и одной или нескольких переменных, каждая из которых имеет неотрицательную целую степень. Математическая запись монома:

|  |  |
| --- | --- |
|  |  |

, где – константа, – переменная,

**Полином** - конечная сумма мономов. Мfтематическая запись полинома:

|  |  |
| --- | --- |
|  |  |

, где – набор всевозможных целых неотрицательных чисел(мультииндекс), - число, (именуемое коэффициент многочлена) зависящее только от мультииндекса *I*.

Наиболее удобным для хранения полиномов, а также для работы с ними, является такая структура данных, как линейный список. Напомню, что **Линейный список** - это структура данных, состоящая из элементов одного типа, связанных между собой последовательно посредством указателей. Каждый элемент списка имеет указатель на следующий элемент. Последний элемент списка указывает на NULL. Элемент, на который нет указателя, является первым элементом списка. Здесь ссылка в каждом узле указывает на следующий узел в списке.

Узлами нашего списка являются мономы, все из которых лексикографически упорядочены между собой по мультииндексу: моном с большим мультииндексом указывает на следующий за ним моном с меньшим индексом. Это гарантирует единственность представления полинома в памяти компьютера.

# Постановка задачи

Выполнение работы предполагает решение следующих задач:

1. Разработка и реализация вспомогательного класса монома (по аналогии с узлом списка) TMonom, а так же класса полинома TPolynom
2. Пример программы, демонстрирующая работу класса TPolynom.
3. Написание набора автоматических тестов с использованием Google C++ Testing Framework и проверка работоспособности методов классов.

# Руководство пользователя

При запуске программы создаются 6 мономов, состоящих из трех переменных с различными коэффициентами и наборами степеней. После чего, с полученными полиномами проводят базовые арифметические операции, а их результат также выводится на экран.

# Руководство программиста

## Описание структуры программы

Программа состоит из следующих модулей:

* Модуль PolinomLib. Статическая библиотека. Включает в себя заголовочный файл Monom.h, в котором описаны методы с реализаций вспомогательного класса *TMonom.* А так же файл Polinom.h, в котором описаны методы с реализацией класса *TPolynom*.
* Модуль PolinomTest. Набор тестов для класса TPolynom и TMonom. Включает в себя файл Polinom*Test.cpp.* Разработаны они с помощью использования Google C++ Testing Framework.
* Модуль Polinom. Пример использования списка. Включает в себя файл с реализацией *main.cpp.*

## Описание структур данных

**Класс TMonom:**

TMonom\* next; - указатель на следующий моном

int \*power; - массив степеней

double coef; - коэффициент

int n; - количество переменных в мономе

TMonom();

TMonom(int \_n, int\* \_power, double \_c);

TMonom(TMonom& A);

virtual ~TMonom();

void SetNext(TMonom\* \_next); - задать адрес следующего монома

void SetPower(int\* \_power); - задать массив степеней монома

void SetC(double \_c); - задать коэффициент при мономе

void SetN(int \_n); - задать количество переменных в мономе

TMonom\* GetNext(); - получить адрес следующего монома

int\* GetPower(); - получить массив степеней монома

double GetC(); - получить коэффициент при мономе

int GetN(); - получить количество переменных в мономе

TMonom operator + (TMonom& mon);

TMonom operator += (TMonom& mon);

TMonom operator - (TMonom& mon);

TMonom operator -= (TMonom& mon);

TMonom operator \* (TMonom& mon);

TMonom operator \*= (TMonom& mon);

TMonom& operator = (TMonom& mon);

bool operator == (TMonom& mon);

bool operator > (TMonom& mon);

bool operator<(TMonom& mon);

friend istream& operator >> (istream& istr, TMonom& mon);

friend ostream& operator << (ostream& ostr, TMonom& mon);

**Класс TPolynom наследуется от класса TMonom:**

TMonom \*start; - указатель на первый моном в полиноме

int n; - количество переменных в каждом мономе

int size; - количество мономов в полиноме

TPolinom(int \_n = 3);

TPolynom(TPolynom& p);

~TPolynom();

Методы:

int GetSize(); - получить число мономов в полиноме

TMonom\* GetStart(); - получить адрес первого монома в полиноме

TPolynom operator + (TPolynom& p);

TPolynom& operator += (TMonom& m);

TPolynom operator - (TPolynom& p);

TPolynom& operator -= (TMonom& m);

TPolynom operator \* (TPolynom& p);

TPolynom& operator = (TPolynom& p);

bool operator == (TPolynom& p);

friend ostream& operator << (ostream& ostr, TPolynom& pol);

## Описание алгоритмов

**Сложение двух полиномов.**

Алгоритм сложения двух полиномов А и В изображен на блок-схеме расположенной ниже.

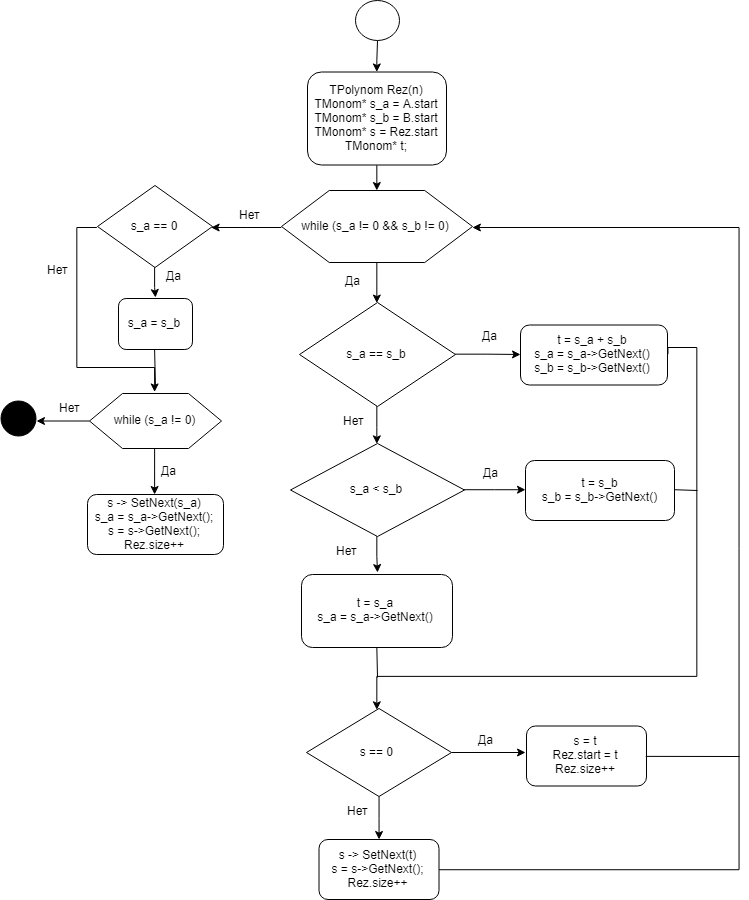


Рисунок 4 Алгоритм сложения двух полиномов

Rez – полином, полученный в результате сложения полиномов А и В, A.start и B.start – адреса первых мономов соответствующих полиномов.

**Сложение полинома и монома.**

Алгоритм сложение полинома и монома:

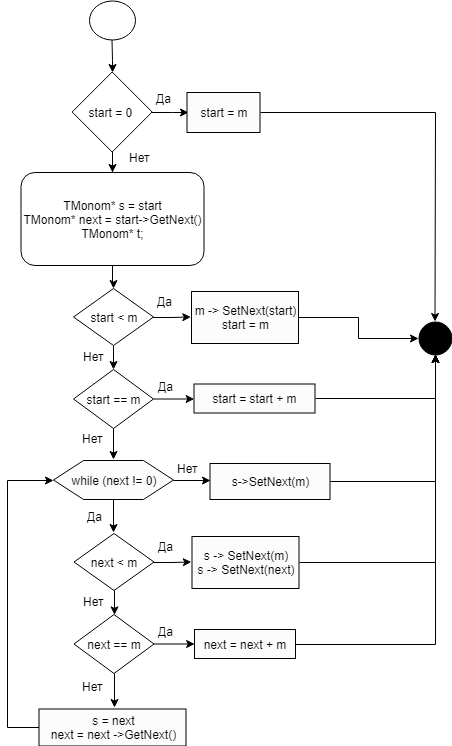


Рисунок 5 Алгоритм сложения полинома и монома

Start – адрес первого монома в полиноме, m – моном.

Алгоритмы нахождения разности и произведения схожи с данными.

# Заключение

В ходе выполнения была разработана система для проведения арифметических операций над многочленами нескольких переменных, реализованная при помощи вспомогательного класс монома TMonom, а также основного класса полинома TPolynom, описаны они в специально разработанной библиотеке PolynomialLib.

Для проверки работоспособности всех методов, были написаны автоматические тесты, реализованные с использованием Google C++ Testing Framework, и пример выполнения арифметических операций.

1. **Литература**
2. Гергель В.П. Методические материалы по курсу «Методы программирования 2»: [http://www.itmm.unn.ru/files/2018/11/Primer-1.7.-Razrabotka-obshhego-predstavleniya-linejnogo-spiska.pdf], 2015.
3. Википедия: свободная электронная энциклопедия: на русском языке: https://ru.wikipedia.org/wiki/Многочлен