

Міністерство освіти і науки України  
Національний технічний університет України «Київський політехнічний  
інститут імені Ігоря Сікорського»  
Факультет інформатики та обчислювальної техніки

Кафедра інформатики та програмної інженерії

Звіт

з лабораторної роботи № 4 з  
дисципліни «Основи програмування -  
2. Методології програмування»

«Перевантаження операторів»

Варіант 21

Виконав студент ІП-13 Макарчук Лідія Олександрівна  
(шифр, прізвище, ім'я, по батькові)

Перевірів Вєчерковська Анастасія Сергіївна  
( прізвище, ім'я, по батькові)

Київ 2022

## Лабораторна робота №4

### Перевантаження операторів

**Мета** – вивчити механізми створення класів з використанням перевантажених операторів (операцій).

Варіант 21

#### Завдання:

21. Визначити клас "Поліном" розмірності 4, членами якого є коефіцієнти полінома. Реалізувати для нього декілька конструкторів, геттери, метод обчислення значення поліному в заданій точці. Перевантажити оператори: "+" / "-" - для збільшення / зменшення усіх коефіцієнтів полінома на вказану величину, "==" – для визначення рівності відповідних коефіцієнтів двох поліномів. Створити три полінома (P1, P2, P3), використовуючи різні конструктори. Коефіцієнти полінома P1 збільшити на вказану величину, а полінома P2 - зменшити на цю ж саму величину. З'ясувати, чи рівні між собою відповідні коефіцієнти поліномів P1 і P2. Обчислити значення поліному P3 в заданій точці.

Виконання завдання мовою C++

#### Код:

//main

```
#include <iostream>
#include <time.h>
#include "Polynomial.h"

using namespace std;

int main()
{
    srand(time(0));
    double c, c0, c1, c2, c3;
    int min, max;
    cout << "Enter one number to initiate coefficients: "; cin >> c;
    init_coefficients(c0, c1, c2, c3);
    init_range(min, max);

    Polynomial P1(c);
    Polynomial P2(c0, c1, c2, c3);
    Polynomial P3(min, max);

    cout << "\nPolynomials\n";
    cout << "P1: " << P1.get_polynomial_as_string();
    cout << "P2: " << P2.get_polynomial_as_string();
    cout << "P3: " << P3.get_polynomial_as_string();

    cout << "\nEnter number to increase/decrease polynomials: ";
    double n; cin >> n;
    P1 += n;
    P2 -= n;

    cout << "\nNew polynomials increased/decreased by " << n << "\n";
```

```

cout << "P1: " << P1.get_polynomial_as_string();
cout << "P2: " << P2.get_polynomial_as_string();

if (P1 == P2)
    cout << "Coefficients of P1 and P2 are equal\n";
else
    cout << "Coefficients of P1 and P2 are NOT equal\n";

cout << "\nEnter x to calculate P3: ";
double x; cin >> x;
cout << "Value of the P3 at the point x = " << P3.calculate_polynomial(x) << "\n";
}

```

## //Polynomial.h

```

#pragma once
#include <string>

using std::string;

class Polynomial
{
    double c0;
    double c1;
    double c2;
    double c3;
    static int generate_number(int rangeMin, int rangeMax);
    static string convert_double_to_string(double n);
public:
    Polynomial(double c = 1);
    Polynomial(double c0, double c1, double c2, double c3);
    Polynomial(int rangeMin, int rangeMax);

    double get_c0();
    double get_c1();
    double get_c2();
    double get_c3();
    string get_polynomial_as_string();
    double calculate_polynomial(double x);

    Polynomial operator+=(double number);
    Polynomial operator-=(double number);
    bool operator==(Polynomial polynomial);
};

void init_coefficients(double& c0, double& c1, double& c2, double& c3);
void init_range(int& rangeMin, int& rangeMax);

```

## //Polynomial.cpp

```

#include "Polynomial.h"
#include <cmath>
#include <iostream>
#include <stdlib.h>

using namespace std;

Polynomial::Polynomial(double c): c0(c), c1(c), c2(c), c3(c) {}
Polynomial::Polynomial(double c0, double c1, double c2, double c3)
{
    this->c0 = c0;
    this->c1 = c1;
    this->c2 = c2;
    this->c3 = c3;
}

```

```

Polynomial::Polynomial(int rangeMin, int rangeMax)
{
    c0 = generate_number(rangeMin, rangeMax);
    c1 = generate_number(rangeMin, rangeMax);
    c2 = generate_number(rangeMin, rangeMax);
    c3 = generate_number(rangeMin, rangeMax);
}
int Polynomial::generate_number(int rangeMin, int rangeMax)
{
    int number = ((double)rand() / RAND_MAX) * (rangeMax - rangeMin) + rangeMin;
    return number;
}
double Polynomial::get_c0() { return c0; }
double Polynomial::get_c1() { return c1; }
double Polynomial::get_c2() { return c2; }
double Polynomial::get_c3() { return c3; }
string Polynomial::convert_double_to_string(double n)
{
    string str = to_string(n);
    size_t pos = str.find_last_not_of('0') + 1;
    str = str.erase(pos);
    size_t lastPos = str.length() - 1;
    if (str[lastPos]=='.')
        str = str.erase(lastPos);
    return str;
}
string Polynomial::get_polynomial_as_string()
{
    string polynomial_string = convert_double_to_string(c0)+ " + " +
convert_double_to_string(c1) + "x + " + convert_double_to_string(c2) + "x^2 + " +
convert_double_to_string(c3) + "x^3" + "\n";
    return polynomial_string;
}
double Polynomial::calculate_polynomial(double x)
{
    return c0 + c1 * x + c2 * pow(x, 2) + c3 * pow(x, 3);
}
Polynomial Polynomial::operator+=(double number)
{
    c0 += number;
    c1 += number;
    c2 += number;
    c3 += number;
    return *this;
}
Polynomial Polynomial::operator-=(double number)
{
    c0 -= number;
    c1 -= number;
    c2 -= number;
    c3 -= number;
    return *this;
}
bool Polynomial::operator==(Polynomial polynomial)
{
    if (c0 == polynomial.c0 && c1 == polynomial.c1 && c2 == polynomial.c2 && c3 ==
polynomial.c3)
        return true;
    return false;
}
void init_coefficients(double& c0, double& c1, double& c2, double& c3)
{
    cout << "Enter 4 numbers to initiate coefficients\n";
    cout << "c0: "; cin >> c0;
    cout << "c1: "; cin >> c1;
    cout << "c2: "; cin >> c2;
    cout << "c3: "; cin >> c3;
}

```

```

}
void init_range(int& rangeMin, int& rangeMax)
{
    cout << "Enter the range for generating coefficients\n";
    cout << "Minimum number of the range: "; cin >> rangeMin;
    cout << "Maximum number of the range: "; cin >> rangeMax;
}

```

## Тестування програми:

 Microsoft Visual Studio Debug Console

```

Enter one number to initiate coefficients: 2.4
Enter 4 numbers to initiate coefficients
c0: 7
c1: -0.79
c2: 5.3
c3: 1
Enter the range for generating coefficients
Minimum number of the range: -5
Maximum number of the range: 5

Polynomials
P1: 2.4 + 2.4x + 2.4x^2 + 2.4x^3
P2: 7 + -0.79x + 5.3x^2 + 1x^3
P3: -2 + 0x + 4x^2 + 4x^3

Enter number to increase/decrease polynomials: 1

New polynomials increased/decreased by 1
P1: 3.4 + 3.4x + 3.4x^2 + 3.4x^3
P2: 6 + -1.79x + 4.3x^2 + 0x^3
Coefficients of P1 and P2 are NOT equal

Enter x to calculate P3: 1
Value of the P3 at the point x = 6

C:\Users\USER1\source\repos\s2laba1cpp\Debug\laba4cpp.exe (process 13828) exited with code 0.
Press any key to close this window . . .

```

Висновок: Під час виконання лабораторної роботи я вивчила механізми створення класів з використанням перевантажених операторів (операцій). У результаті я розробила клас «Поліном розмірності 4» та створила 3 об'єкти даного класу: P1, P2, P3. За допомогою перевантажених операторів скороченого присвоєння я збільшила значення коефіцієнтів P1 та зменшила значення коефіцієнтів P2 на одну і ту ж саму величину. Після цього за допомогою перевантаженого оператора порівняння порівняла відповідні коефіцієнти P1 та P2. Також я знайшла значення поліному P3 у вказаній точці x.