МИНЕСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

УЧЕРЕЖДЕНИЕ ОБРАЗОВАНИЯ

«Брестский государственный технический университет»

Кафедра «Интеллектуальные информационные технологии»

Лабораторная работа №3

По дисциплине «Вычислительная математика»

За 2 семестр

Тема: «Интерполирование функций полиномом»

Выполнила:

студентка 1 курса

группы АС-56

Карпенко М.В.

Проверил:

Пролиско Е.Е.

Брест 2020

Вариант 29

Задана функция y = -3.8\*x-4.3\*cos^2(0.9\*x+2)+2.6 на интервале [-6.9, -0.9].

1. Построить полиномы Лагранжа и Ньютона на равномерной сетке. Количество узлов интерполирования n=8

2. Вычислить реальную абсолютную погрешность интерполирования в точке x=-2.7

*Код программы (полином Ньютона):*

#include <iostream>

#include<cmath>

using namespace std;

double f(double x) //возвращает значение функции в данной точке

{

return -3.8 \* x - 4.3 \* pow(cos(0.9 \* x + 2), 2) + 2.6;

}

int main()

{

const int size = 8;

double data[2 \* size][size] = { 0 };

double datax[size];

double a = -6.9, b = -0.9;

double l = (b - a) / (size - 1);

int q = 0, qq = 0; //счетчики

for (int i = 0; i < size; i++)//внесение в таблицу начальные данные

{

datax[qq] = a;

data[q][0] = f(a);

q += 2;

qq++;

a += l;

}

int x = 0;

for (int i = 1; i < size; i++)//заполнение таблицы

{

for (int j = 0, k = 0; j < (size \* 2); j += 2, k++)

{

double temp = datax[k + x + 1] - datax[k];

data[i + j][i] = (data[i + j + 1][i - 1] - data[i + j - 1][i - 1]) / temp;

}

x++;

}

double X = -2.7;//искомая точка

double rar[size - 1];

double ar[size - 1];

for (int i = 0; i < size - 1; i++)

{

rar[i] = X - datax[i]; //вычисление всех скобок (х-)

ar[i] = 1;

}

for (int i = 0; i < size - 1; i++)

{

for (int j = 0; j <= i; j++)

{

ar[i] \*= rar[j]; //вычисление коэффициентов

}

}

double result = data[0][0];

for (int i = 1; i < size - 1; i++)

{

result += (data[i][i] \* ar[i - 1]);

}

cout << result << endl;

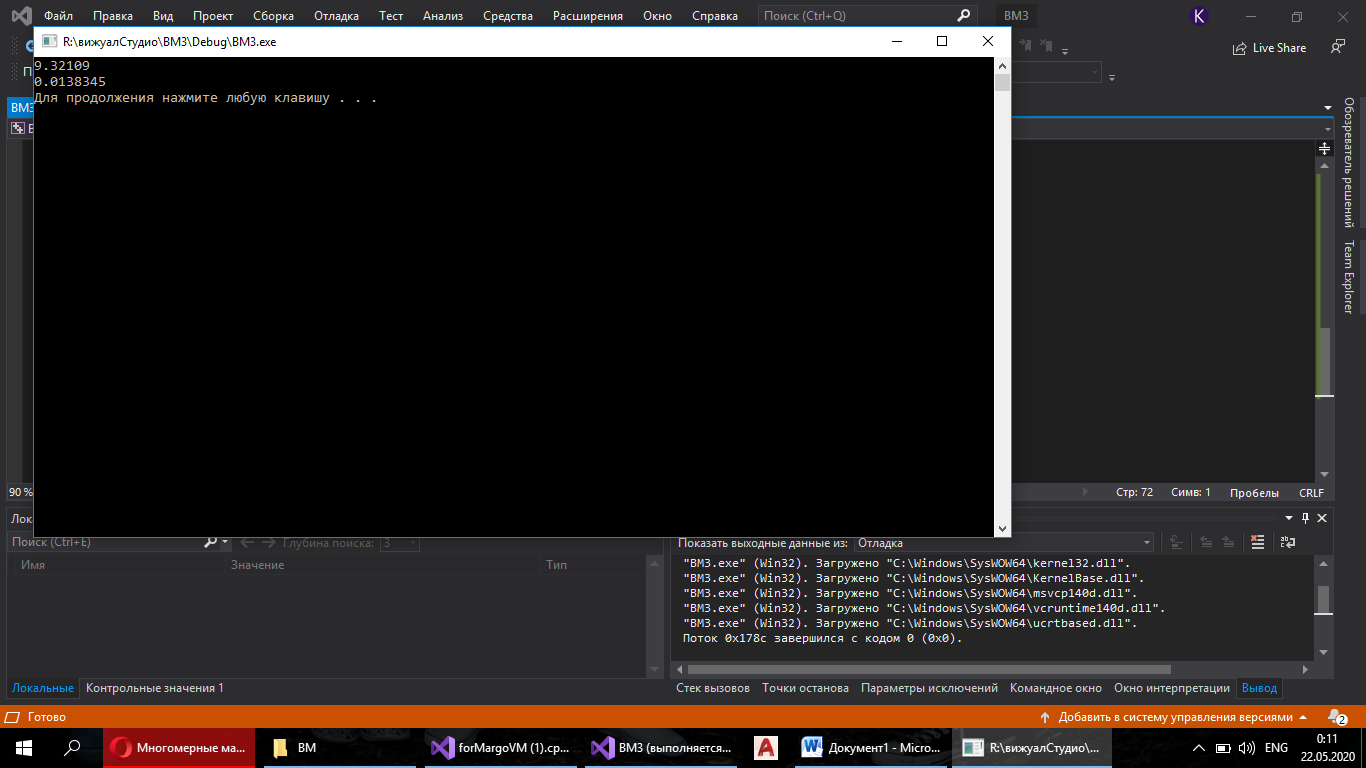
cout << abs(result - f(X)) << endl; //невязка

system("pause");

return 0;

}

*Результат работы программы:*



*Код программы (полином Лагранжа):*

#include <iostream>

#include<cmath>

using namespace std;

double InterpolateLagrangePolynomial(double x, double\* x\_values, double\* y\_values, int size)

{

double lagrange\_pol = 0;

double basics\_pol;

for (int i = 0; i < size; i++)

{

basics\_pol = 1;

for (int j = 0; j < size; j++)

{

if (j == i) continue;

basics\_pol \*= (x - x\_values[j]) / (x\_values[i] - x\_values[j]);

}

lagrange\_pol += basics\_pol \* y\_values[i];

}

return lagrange\_pol;

}

double testF(double x)

{

return -3.8 \* x - 4.3 \* pow(cos(0.9 \* x + 2), 2) + 2.6;

}

int main()

{

const int size = 7;

double r = (-0.9 + 6.9) / size;

double x\_values[size];

double y\_values[size];

int j = 0;

for (double i = -6.9; i < -0.9; i += r) //заполнение массивов значениями

{

x\_values[j] = i;

y\_values[j] = testF(i);

j++;

}

cout << InterpolateLagrangePolynomial(-2.7, x\_values, y\_values, size) << endl;

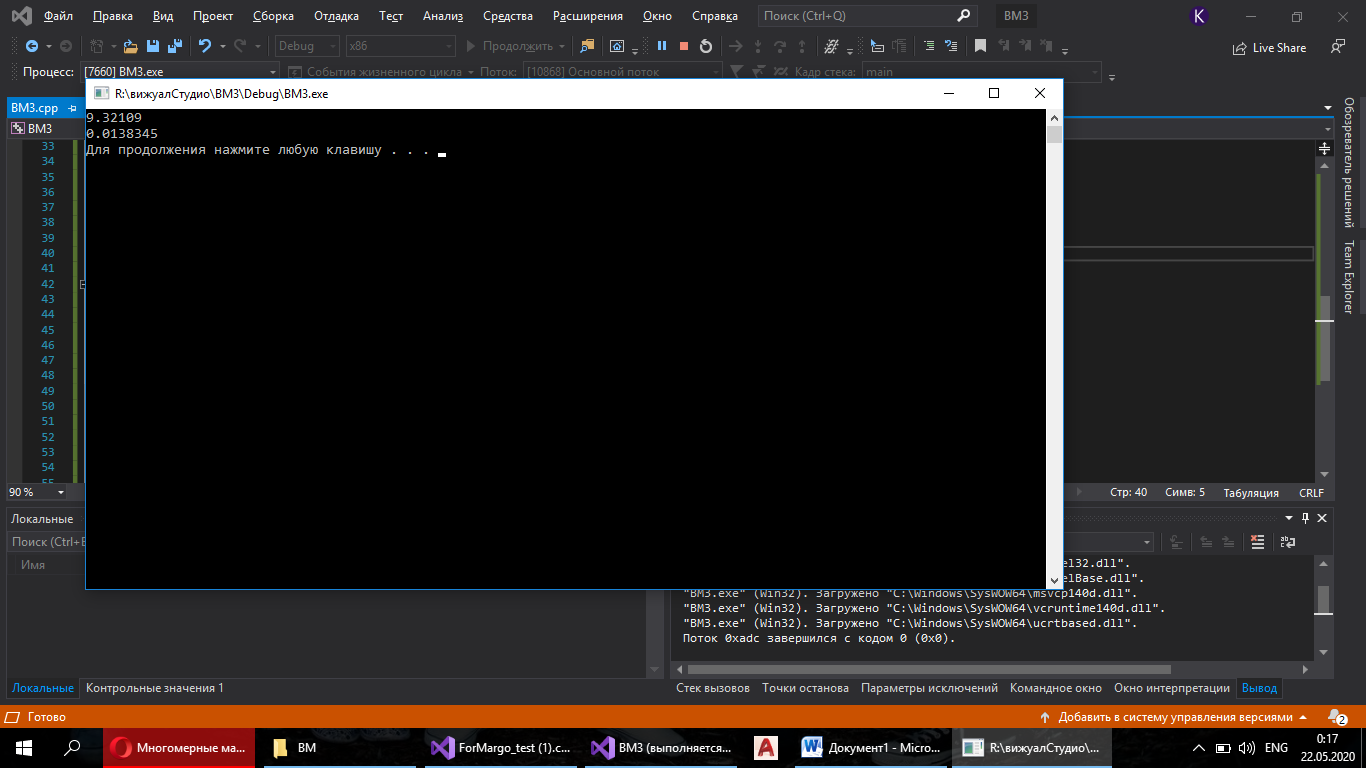
cout << abs(InterpolateLagrangePolynomial(-2.7, x\_values, y\_values, size)-testF(-2.7)) << endl;

system("pause");

return 0;

}

*Результат работы программы:*



*Вывод :*  Разработала программы интерполирования функций полиномами Ньютона и Лагранжа. Значения в обеих программах сошлись.