**ЛАБОРАТОРНАЯ РАБОТА №3**

Аппроксимация функций. Метод наименьших квадратов

**Цель работы**: изучить метод наименьших квадратов, линейную и квадратичную аппроксимации, алгоритм метода, формулы для вычисления, написать программу на языке программирования, реализующую линейную и квадратичную аппроксимации методом наименьших квадратов.

Вариант 3

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

ЗАДАНИЕ 1

***Ход решения для линейной аппроксимации*:**

1. Определить по описанным формулам.
2. Вычислить .
3. Задать любое значение и вывести на экран результаты: аппроксимирующую линейную функцию *.*

Для начала нужно определить .

; ; ; ;

Далее вычислим .

КОД ПРОГРАММЫ

class Program

{

//функция для линейной аппроксимации

static double fi(double a0, double a1, double x)

{

return a0 + a1 \* x;

}

static void Main(string[] args)

{

//создание массива

int n = 8;

double[] x = new double[8]{4, 9, 19, 25.7, 33.2, 40.7, 48.2, 55.7 };

double[] y = new double[8] { 2.4, 5.8, 6.4, 8.9, 10.9, 12.9, 14.9, 16.9 };

Console.WriteLine("Линейная аппроксимация: \n");

double s1 = 0, s2 = 0, s3 = 0, s4 = 0;

//вычисление s1-s4

for (int i = 0; i < x.Length; i++)

{

s1 += x[i];

s3 += Math.Pow(x[i], 2);

}

for (int i = 0; i < y.Length; i++)

{

s2 += y[i];

s4 += x[i] \* y[i];

}

//вычисление а0 и a1

double a0 = (s2 \* s3 - s1 \* s4) / (n \* s3 - Math.Pow(s1, 2));

double a1 = (n \* s4 - s1 \* s2) / (n \* s3 - Math.Pow(s1, 2));

//вывод на экран вычисленных значений

Console.WriteLine("s1= {0}, s2= {1}, s3= {2}, s4= {3}", s1, s2, s3, s4);

Console.WriteLine("\na0= {0},\na1= {1}\n", a0, a1);

//вывод на экран значения функции

for(int i = 0; i < x.Length; i++)

{

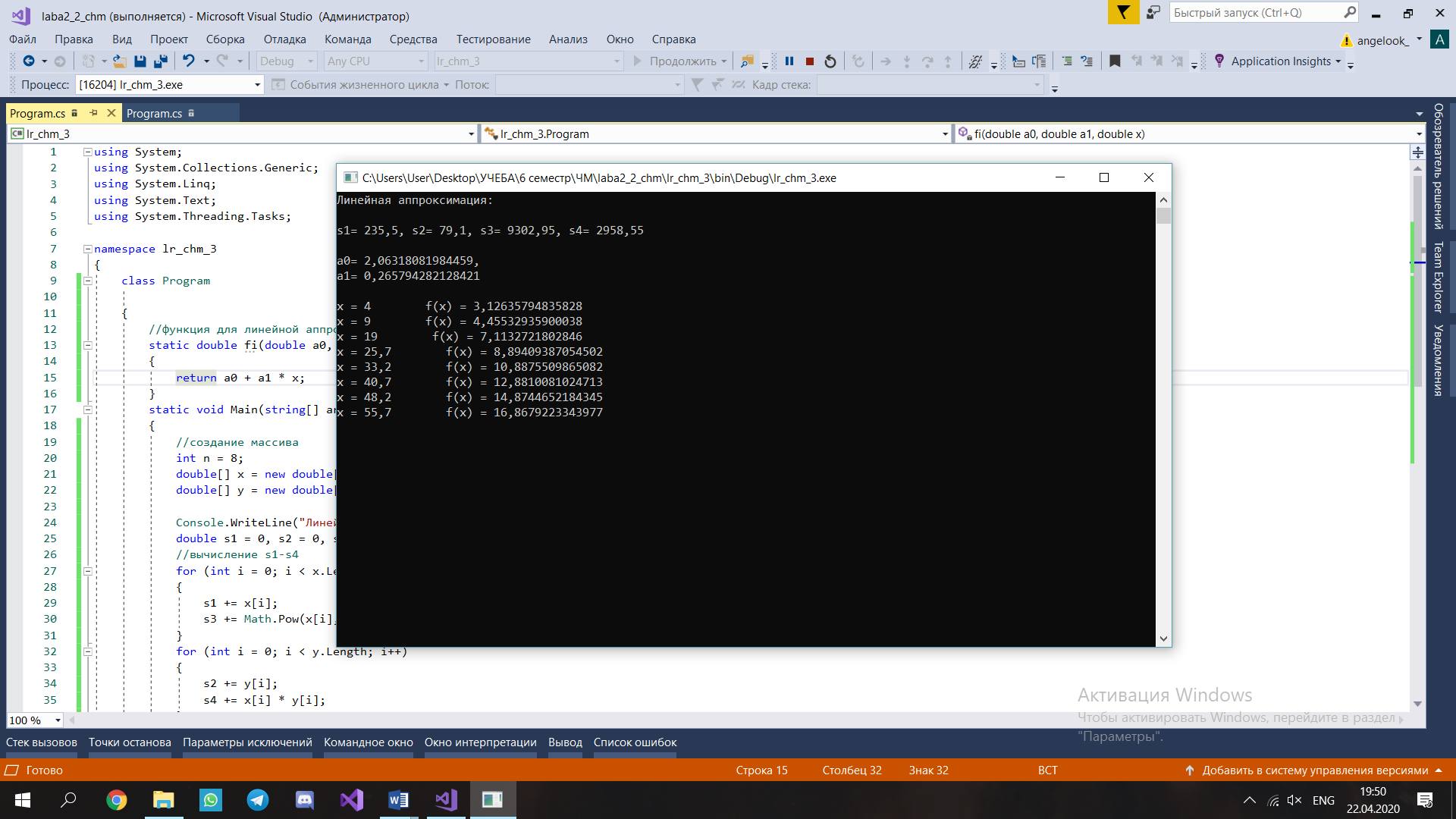
Console.WriteLine("x = " + x[i] + " f(x) = " + fi(a0, a1, x[i]));

}

Console.ReadKey();

}

ВЫВОД НА ЭКРАН



Представлены решение и проверка функции.

ЗАДАНИЕ 2

***Ход решения для квадратичной аппроксимации:***

1. Определить по описанным формулам.
2. Вычислить .
3. Задать любое значение и вывести на экран результаты: аппроксимирующую квадратичную функцию.

Для начала нужно определить .

; ; ; ;

Далее вычислим .

КОД ПРОГРАММЫ

class Program

{

static double delta(double c1, double c2, double c4, double c6)

{

return 8 \* (c2 \* c6 - Math.Pow(c4, 2)) - c1 \* (c1 \* c6 - c2 \* c4)

+ c2 \*(c1 \* c4 - Math.Pow(c2, 2));

}

static double delta1(double c1, double c2, double c3, double c4, double c5, double c6, double c7)

{

return (c3 \* c2 \* c6 + c5 \* c4 \* c2 + c7 \* c1 \* c4 - c2 \* c2 \* c7

- c4 \* c4 \*c3 - c6 \* c1 \* c5);

}

static double delta2(double c1, double c2, double c3, double c4, double c5, double c6, double c7) //вычисление ∆2

{

return (8 \* c5 \* c6 - 8 \* c7 \* c4 - c3 \* c1 \* c6 + c3 \* c2 \* c4 + c2 \* c1 \* c7 - c2 \* c5 \* c2);

}

static double delta3(double c1, double c2, double c3, double c4, double c5, double c6, double c7) //вычисление ∆3

{

return (8 \* c2 \* c7 - 8 \* c5 \* c4 - c1 \* c1 \* c7 + c1 \* c2 \* c5 + c3 \* c1 \* c4 - c2 \* c3 \* c2);

}

//функция для квадратичной аппроксимации

static double fi(double a0, double a1, double a2, double x)

{

return a0 + a1 \* x + a2\*Math.Pow(x,2);

}

static void Main(string[] args)

{

//создание массива

int n = 8;

double[] x = new double[8]{4, 9, 19, 25.7, 33.2, 40.7, 48.2, 55.7 };

double[] y = new double[8] { 2.4, 5.8, 6.4, 8.9, 10.9, 12.9, 14.9, 16.9 };

Console.WriteLine("Квадратичная аппроксимация: \n");

double s1 = 0, s2 = 0, s3 = 0, s4 = 0;

//вычисление s1-s4

for (int i = 0; i < x.Length; i++)

{

s1 += x[i];

s3 += Math.Pow(x[i], 2);

}

for (int i = 0; i < y.Length; i++)

{

s2 += y[i];

s4 += x[i] \* y[i];

}

double c1 = 0, c6 = 0, c7 = 0, c4 = 0, c2=0,c3=0,c5=0;

for (int i = 0; i<x.Length; i++)

{

c4 += Math.Pow(x[i], 3);

c6 += Math.Pow(x[i], 4);

c7 = c7 + (x[i] \* x[i] \* y[i]);

}

//вычисление а0 и a1

double a0 = delta1(c1, c2, c3, c4, c5, c6, c7) / delta(c1, c2, c4, c6);

double a1 = delta2(c1, c2, c3, c4, c5, c6, c7) / delta(c1, c2, c4, c6);

double a2 = delta3(c1, c2, c3, c4, c5, c6, c7) / delta(c1, c2, c4, c6);

//вывод на экран вычисленных значений

Console.WriteLine("s1= {0}, s2= {1}, s3= {2}, s4= {3}", s1, s2, s3, s4);

Console.WriteLine("с4= {0}, с5= {1}, с6= {2} ", c4, c6, c7);

Console.WriteLine("\na0= {0},\na1= {1}, \na2={2}", a0, a1,a2);

//вывод на экран значения функции

for(int i = 0; i < x.Length; i++)

{

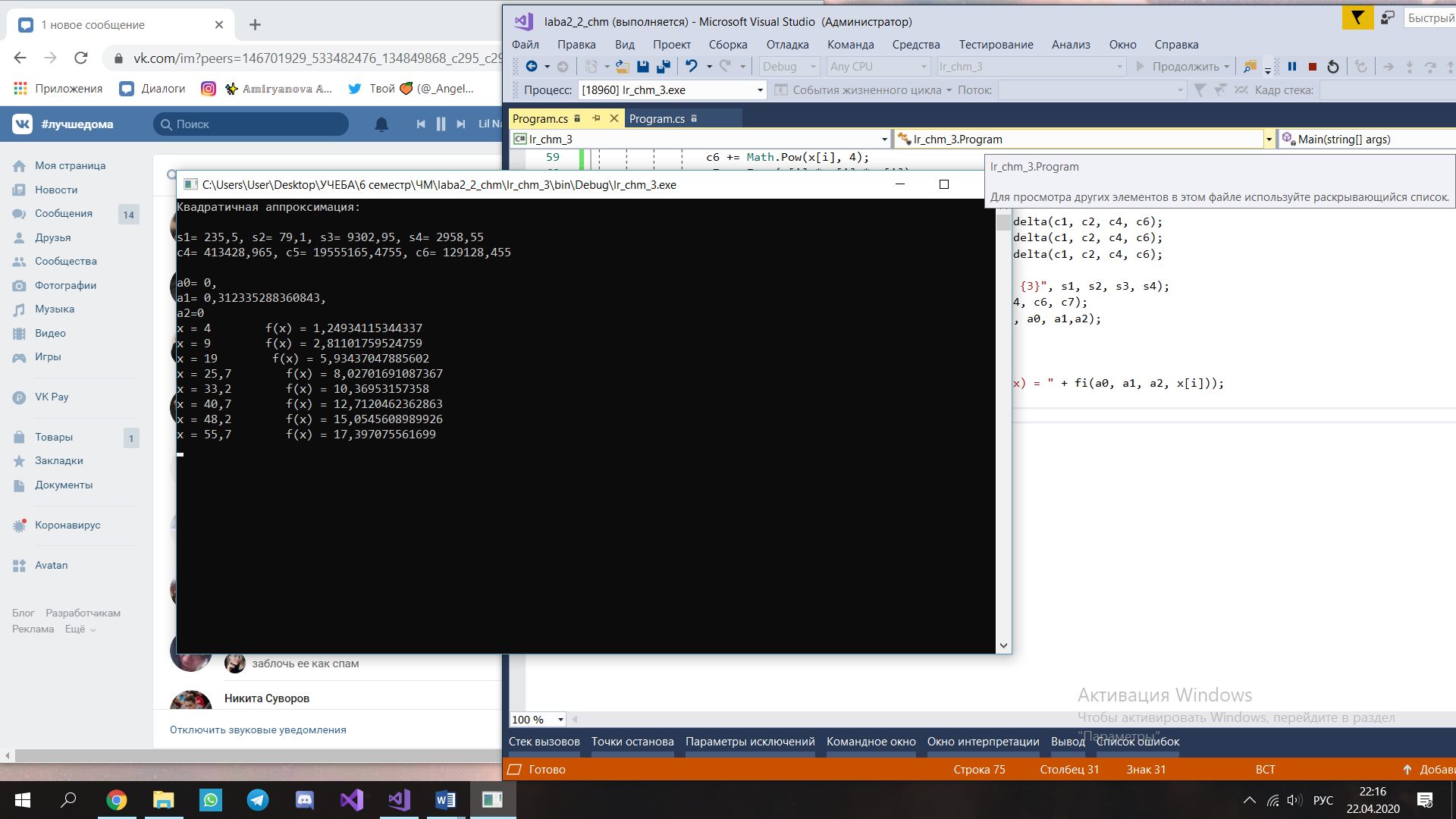
Console.WriteLine("x = " + x[i] + " f(x) = " + fi(a0, a1, a2, x[i]));

}

Console.ReadKey();

}

ВЫВОД НА ЭКРАН



Представлены решение и проверка.