

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Казанский национальный исследовательский технологический университет»

(ФГБОУ ВО «КНИТУ»)

Институт Нефти, химии и нанотехнологий

Факультет Наноматериалов и нанотехнологий

**Лабораторная работа по дисциплине  
 «Методы оптимизации»**

на тему: «Метод дихотомии»

**Выполнил**: студент группы 4381-22, Тимушев Даниил Анатольевич

**Проверил:** ДТН, профессор кафедры ИСУИР Герасимов Александр Викторович

Казань 2021

ПОСТАНОВКА ЗАДАЧИ

Метод относится к последовательным стратегиям. Задается начальный интервал неопределенности и требуемая точность. Алгоритм опирается на анализ значений функции в двух точках. Для их нахождения текущий интервал неопределенности делится пополам. Поиск заканчивается, если длина текущего интервала неопределенности меньше заданной величины.

Алгоритм поиска минимума функции сводится к выполнению следующих этапов:

1. Задается начальный интервал неопределенности L0 = [a0, b0] и p – заданная погрешность вычисления xm.

2. Проверяется условие |b - a| < 2p. Если это условие выполняется, то идем к п.7, иначе выполняется п.3.

3. Интервал поиска [a, b] делится пополам и вычисляются две абсциссы, симметрично расположенные относительно точки x = (a + b) / 2: x1 = (a + b - p) / 2 и x2 = (a + b + p) / 2.

4. Для этих значений х вычисляем F(x1) и F(x2).

5. Проверяется условие F(x1) > F(x2). Если условие выполняется, то полагаем, что b = x2 и идем к п.2, иначе идем к п.6.

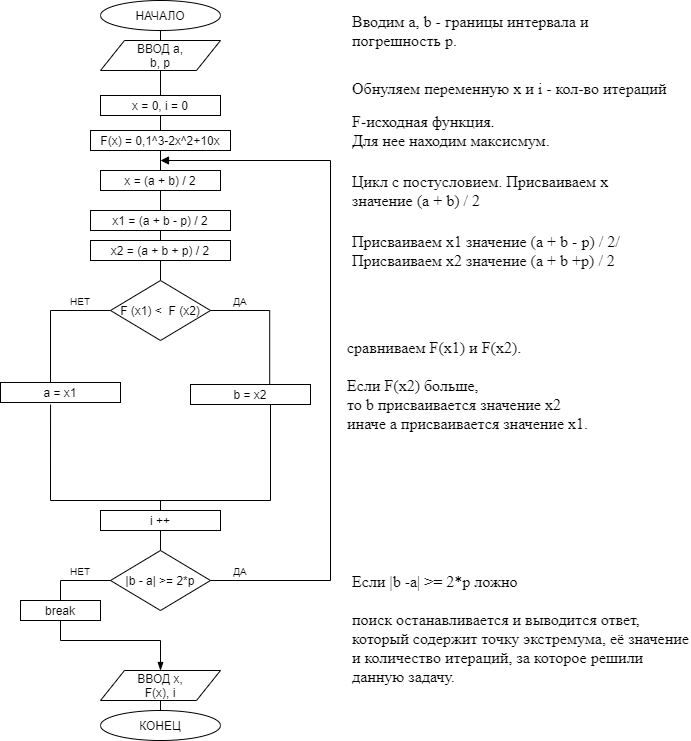
6. Полагаем, что а = x1 и идем к п.2.

7. Выводится на печать xm и F(xm).

Для оценки сходимости метода используется характеристика относительного уменьшения начального интервала неопределенности количество вычислений функции.

Требуется найти максимум функции 0,1*x*^3-2*x*^2+10*x*.

АЛГОРИТМ



КОД ПРОГРАММЫ

Программа на языке C# для реализации поиска максимума методом дихотомии.

Для удобной работы с функцией, для которой нужно найти максимум, создадим метод func, возвращающий значение данной нам функции для аргумента.

И введем значения для a, b и p - погрешность.

Создаются точка х и переменные х1 и х2.

public static double func(double x)

{

return 0.1 \* Math.Pow(x, 3) - 2 \* Math.Pow(x, 2) + 10 \* x;

}

double i, a, b, p;

private void textBox1\_TextChanged(object sender, EventArgs e)

{

a = Convert.ToDouble(textBox1.Text);

}

private void textBox3\_TextChanged(object sender, EventArgs e)

{

b = Convert.ToDouble(textBox3.Text);

}

private void textBox7\_TextChanged(object sender, EventArgs e)

{

p = Convert.ToDouble(textBox7.Text);

}

double x = 0;

double x1, x2;

Далее начинается цикл с постусловием |b -a| >= 2\*p , в котором x присваиваем значение (a + b) / 2, x1 присваиваем значение (a + b - p) / 2, x2 присваиваем значение (a + b + p) / 2. Сравниваем func (x1) и func(x2). Если func (x1) меньше, то b присваиваем значение х2, иначе а присваиваем значение х1, попутно увеличивая количество итераций процесса поиска максимума.

while (Math.Abs (b -a)>=2\*p)

{

x = (a + b) / 2;

x1 = (a + b - p) / 2;

x2 = (a + b + p) / 2;

if (func(x1) < func(x2))

b = x2;

else a = x1;

i++;

textBox4.Text += $"x = {x}, Func(x) = {func(x)}, I = {i}" + Environment.NewLine;

}

textBox2.Text = x.ToString();

textBox5.Text = func(x).ToString();

textBox6.Text = i.ToString();

Если |b -a| станет меньше 2\*p, заканчивается цикл и выводим ответ, который содержит точку экстремума, её значение и количество итераций.

РЕЗУЛЬТАТЫ ПРОГРАММЫ

Было создано приложение Windows Forms для визуализации программы. На рисунке 1 представлена форма для реализации метода равномерного поиска.

Здесь имеются элементы TextBox для записи в них начального значения и шага. После ввода данных при нажатии кнопки «ВЫЧИСЛИТЬ» в окошке результатов появятся значения:

« х = » - здесь записывается значения экстремума (точки, в которой функция принимает максимальное значение);

« F(x) = » - здесь записывается максимальное значение функции;

« i = » - здесь отображается количество итераций, за которое программа нашла решение.

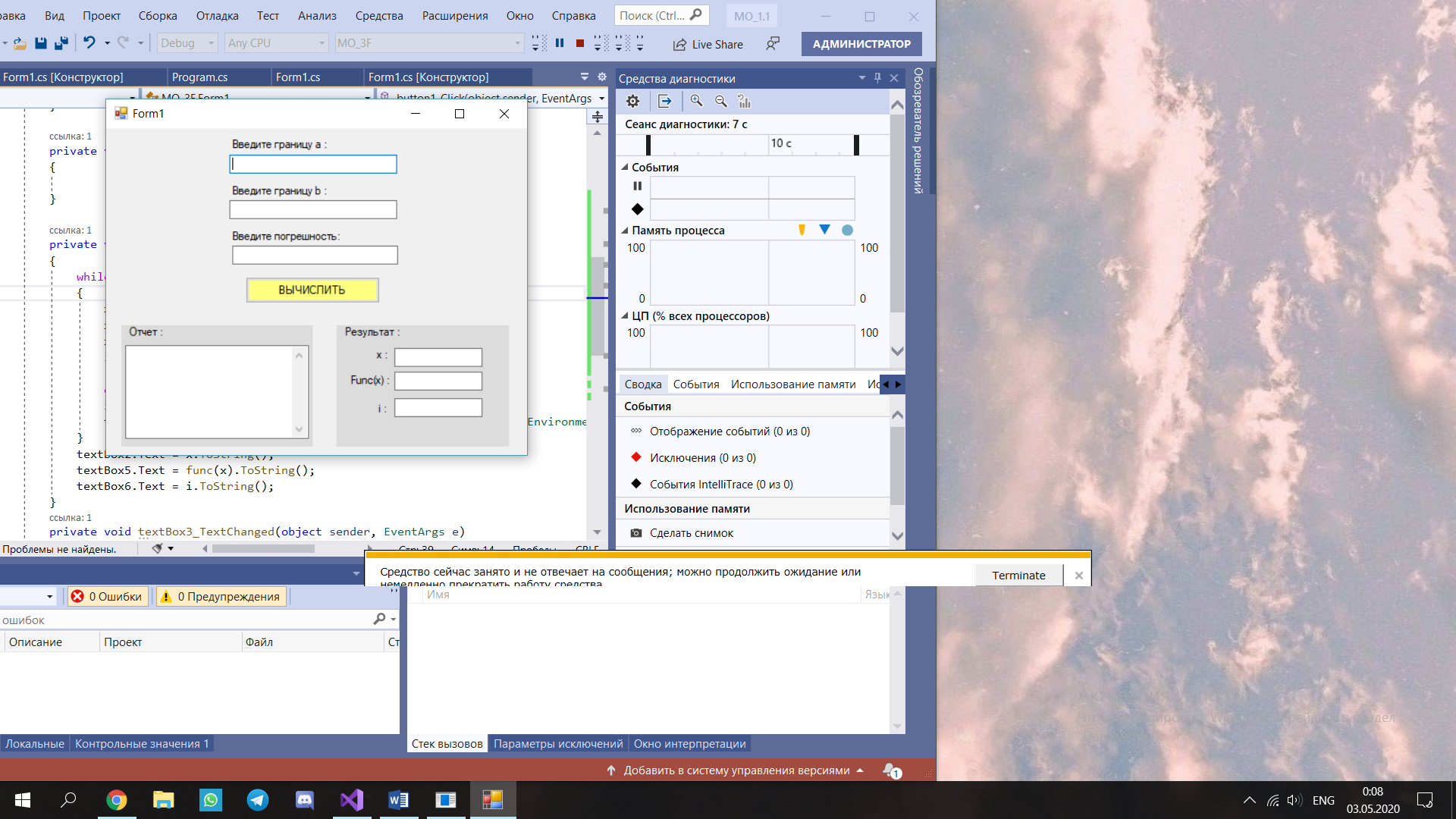


Рис. 1 – Форма

В окошке «Отчет» будет появляться значения *x*, func(*x*), i.

Введем границы интервала 1 и 4, погрешность 0,0001 увидим результат выполнения программы на рисунке 2.   
За 15 итерации был найден максимум равный 8,1008917289390 в

*х* =1,0001415496826.

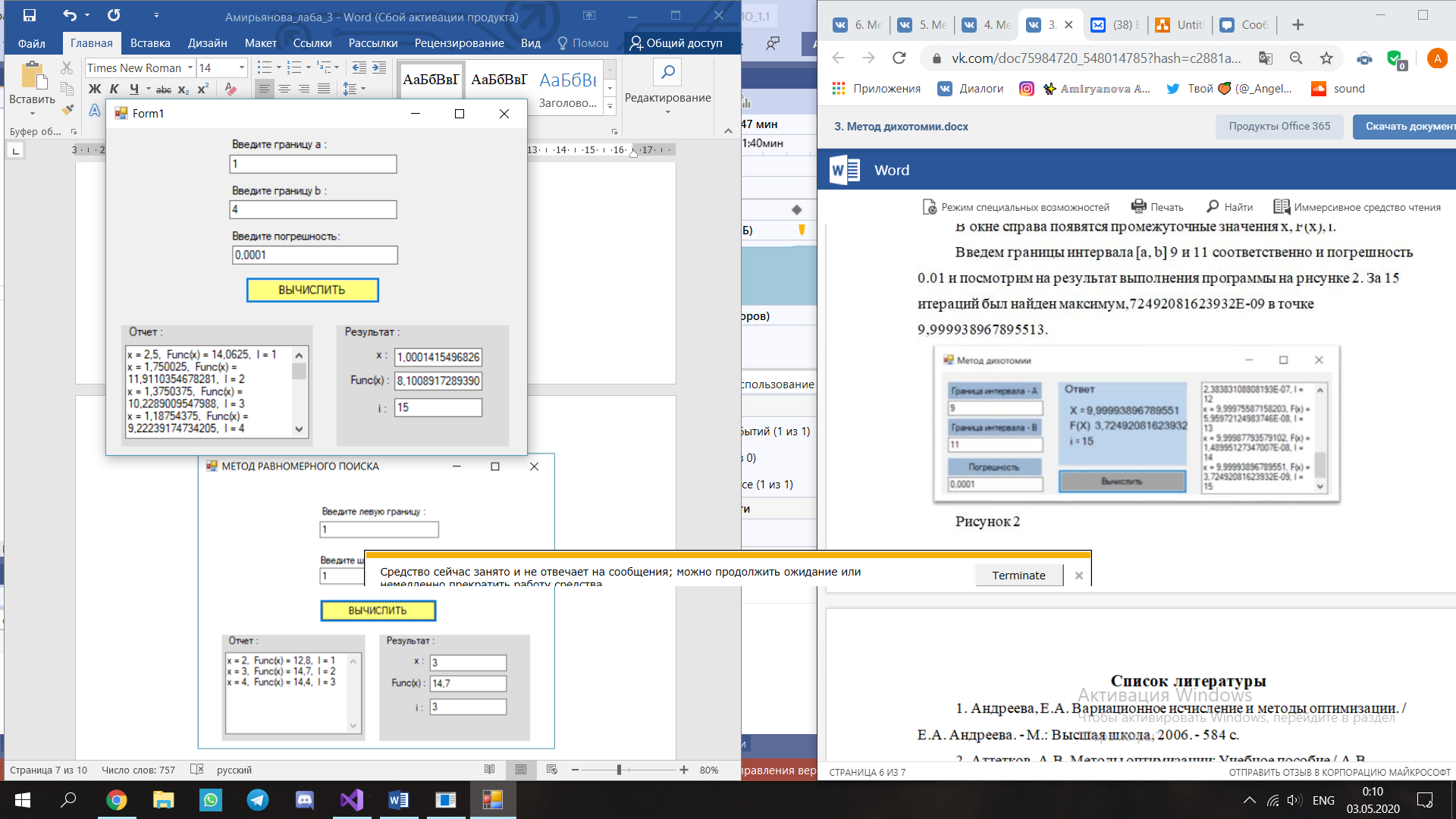


Рис. 2 – Результаты тестирования

СПИСОК ЛИТЕРАТУРЫ

1. Бродецкий, Г.Л. Экономико-математические методы и модели в логистике: процедуры оптимизации: Учеб. для студентов учреждений высшего профессионального образования / Г.Л. Бродецкий. - М.: ИЦ Академия, 2012.
2. Аттетков, А.В. Методы оптимизации: Учебное пособие / А.В. Аттетков, В.С. Зарубин, А.Н.Канатников. - М.: ИЦ РИОР, НИЦ Инфра-М, 2013.
3. Васильев, Ф.П. Методы оптимизации в 2-х книгах. Кн.2 / Ф.П. Васильев. - М.: МЦНМО, 2011.
4. Аттетков, А.В. Введение в методы оптимизации / А.В. Аттетков, В.С. Зарубин, А.Н. Канатников. - М.: Финансы и статистика, 2008.
5. Андреева, Е.А. Вариационное исчисление и методы оптимизации. / Е.А. Андреева. - М.: Высшая школа, 2006.