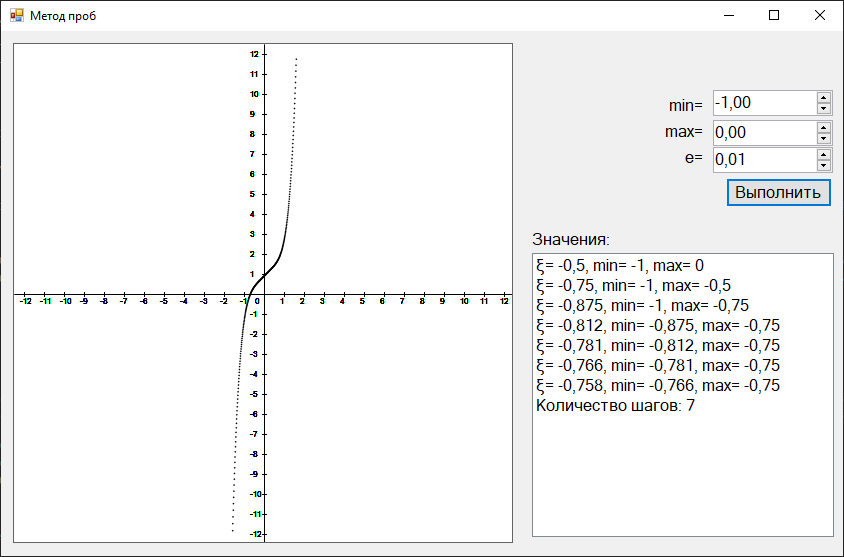
Практическая работа №1 «Метод половинного деления»

Цель работы:

Умение разрабатывать алгоритмы и программы для решения вычислительных задач, учитывая необходимую точность получаемого результата;

Уравнение: 

Результат работы программы:



Листнинг:

namespace mathlab

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

private double min, max, err;

public double func(double x)

{

double result = 0;

try

{

result = Math.Pow(x, 5) + x + 1; // Math.Pow(x, 3) + 4 \* x - 6; //

}

catch (Exception e)

{

MessageBox.Show(e.Message);

}

return result;

}

public void CalculateFirst()

{

listBox1.Items.Clear();

int iter = 0;

try

{

while (Math.Abs(max - min) >= err)

{

iter++;

double x = (min + max) / 2;

listBox1.Items.Add($"ξ= {Math.Round(x, 3)}, min= {Math.Round(min, 3)}, max= {Math.Round(max, 3)}");

if (func(x) \* func(min) > 0)

{

min = x;

}

else

{

max = x;

}

}

}

catch (Exception e)

{

MessageBox.Show(e.Message);

}

listBox1.Items.Add($"Количество шагов: {iter}");

}

private void DrawGraph()

{

Graphics g = pictureBox1.CreateGraphics();

g.SmoothingMode = System.Drawing.Drawing2D.SmoothingMode.AntiAlias;

Pen pen = new Pen(Color.Black);

Point x1, x2, y1, y2;

int width, height;

width = pictureBox1.Width;

height = pictureBox1.Height;

x1 = new Point(0, height / 2);

x2 = new Point(width, height / 2);

y1 = new Point(width / 2, 0);

y2 = new Point(width / 2, height);

g.DrawLine(pen, x1, x2);

g.DrawLine(pen, y1, y2);

for (double fx = -20; fx < 20; fx += 0.01)

{

if (fx - Math.Round(fx) > 0.00000000000001)

{

int xleft = (int)fx \* 20 + width / 2;

g.DrawLine(pen, new Point(xleft, height / 2 - 2), new Point(xleft, height / 2 + 2));

g.DrawLine(pen, new Point(width / 2 - 2, xleft), new Point(width / 2 + 2, xleft));

double axys\_coords = (double)Math.Round(fx);

if (Math.Round(axys\_coords) < 0)

{

axys\_coords += 1;

}

if (Math.Round(axys\_coords) == 0) g.DrawString(0.ToString(), new Font("Arial", 6), new SolidBrush(Color.Black), new Point(xleft - 10, height / 2 + 2));

else

{

g.DrawString(Math.Round(axys\_coords).ToString(), new Font("Arial", 6), new SolidBrush(Color.Black), new Point(xleft - 5, height / 2 + 2));

g.DrawString(Math.Round(-axys\_coords).ToString(), new Font("Arial", 6), new SolidBrush(Color.Black), new Point(height / 2 - 15, xleft - 5));

}

}

try

{

double fy = func(fx);

if (fy > 12 || fy < -12) { continue; }

else

{

float xx = (float)(fx \* 20f);

float yy = (float)(-fy \* 20f);

g.DrawEllipse(pen, xx + width / 2, yy + height / 2, 1, 1);

}

}

catch (Exception e)

{

MessageBox.Show(e.Message);

}

}

}

private void button1\_Click(object sender, EventArgs e)

{

min = (double)numericUpDown1.Value;

max = (double)numericUpDown2.Value;

err = (double)numericUpDown3.Value;

CalculateFirst();

DrawGraph();

}

}

}