# Systems Of Equations

## Screenshots

A screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

## Code

namespace fin1

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

private void Form1\_Load(object sender, EventArgs e)

{

}

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

{

//create an array for both lines.

double[] lineA = new double[3];

double[] lineB = new double[3];

if (inputBox.SelectedIndex == 0) // coeff input

{

//convert to double

lineA[0] = double.Parse(lineAtextBox1.Text);

lineA[1] = double.Parse(lineAtextBox2.Text);

lineA[2] = double.Parse(lineAtextBox3.Text);

lineB[0] = double.Parse(lineBtextBox1.Text);

lineB[1] = double.Parse(lineBtextBox2.Text);

lineB[2] = double.Parse(lineBtextBox3.Text);

}

else if (inputBox.SelectedIndex == 1) // file input

{

string filePath = fileTextBox.Text;

//take the file and convert them to solve

if (!File.Exists(filePath))

{

MessageBox.Show("File not found!", "Error", MessageBoxButtons.OK, MessageBoxIcon.Error);

return;

}

try

{

string[] lines = File.ReadAllLines(filePath);

if (lines.Length >= 2)

{

//split the text file

string[] lineACoeffs = lines[0].Split(',');

string[] lineBCoeffs = lines[1].Split(',');

//convert to doubles

lineA[0] = double.Parse(lineACoeffs[0]);

lineA[1] = double.Parse(lineACoeffs[1]);

lineA[2] = double.Parse(lineACoeffs[2]);

lineB[0] = double.Parse(lineBCoeffs[0]);

lineB[1] = double.Parse(lineBCoeffs[1]);

lineB[2] = double.Parse(lineBCoeffs[2]);

}

else

{

/\* file has to be in the format of

\* x, y, z

\* a, b, c

\*

\*/

MessageBox.Show("Invalid file format!", "Error", MessageBoxButtons.OK, MessageBoxIcon.Error);

return;

}

}

catch (Exception ex)

{

MessageBox.Show("Error reading file: " + ex.Message, "Error", MessageBoxButtons.OK, MessageBoxIcon.Error);

return;

}

}

// solve the system of equations

//find the determinate

double d = lineA[0] \* lineB[1] - lineA[1] \* lineB[0];

if (d == 0)

{

//determine if both lines are the same or paralell

if (lineA[2] == lineB[2])

{

resultLabel.Text = "The two lines are the same line.";

}

else

{

resultLabel.Text = "The two lines are parallel and do not intersect.";

}

}

else

{

double x = (lineA[1] \* lineA[2] - lineA[1] \* lineB[2]) / d;

double y = (lineB[0] \* lineB[2] - lineB[0] \* lineA[2]) / d;

resultLabel.Text = $"Intersection point: ({x}, {y})";

}

}

private void inputTypeComboBox\_SelectedIndexChanged(object sender, EventArgs e)

{

if (inputBox.SelectedIndex == 0) // coefficients input

{

coeffBox.Enabled = true;

fileBox.Enabled = false;

}

else if (inputBox.SelectedIndex == 1) // file input

{

coeffBox.Enabled = false;

fileBox.Enabled = true;

}

}

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

{

//create the file streaming.

OpenFileDialog openFileDialog = new OpenFileDialog();

openFileDialog.Filter = "Text Files (\*.txt)|\*.txt";

openFileDialog.Title = "Select Coefficients File";

if (openFileDialog.ShowDialog() == DialogResult.OK)

{

fileTextBox.Text = openFileDialog.FileName;

}

}

}

}