Problem 1

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace Problem1

{

public partial class Form1 : Form

{

public Form1()

{

this.BackColor = Color.FromArgb(0, 255, 0);

this.MouseMove += MyMouseMove;

}

private void MyMouseMove(object sender, MouseEventArgs e)

{

if (e.X < (this.ClientSize.Width) / 4)

{

this.Show();

this.Location = new Point(this.Location.X, this.Location.Y);

this.BackColor = Color.FromArgb(0, 255, 0);

}

if ((e.X > (this.ClientSize.Width) / 4) && (e.X < (this.ClientSize.Width) / 2))

{

this.Show();

this.Location = new Point(this.Location.X, this.Location.Y);

this.BackColor = Color.FromArgb(255, 0, 0);

}

if ((e.X > (this.ClientSize.Width) / 2) && (e.X < (this.ClientSize.Width) \* 3 / 4))

{

this.Show();

this.Location = new Point(this.Location.X, this.Location.Y);

this.BackColor = Color.FromArgb(0, 0, 255);

}

if (e.X > (this.ClientSize.Width) \* 3 / 4)

{

this.Show();

this.Location = new Point(this.Location.X, this.Location.Y);

this.BackColor = Color.FromArgb(0, 0, 0);

}

}

}

}

Problem 2

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace Problem2

{

public partial class Form1 : Form

{

int ct = 0;

int num = 9;

double opacity;

public Form1()

{

this.BackColor = Color.FromArgb(0, 0, 0);

this.MouseDown += MyMouseDown;

}

private void MyMouseDown(object sender, MouseEventArgs e)

{

ct++;

if (ct < num)

{

opacity = this.Opacity - 0.1;

this.Opacity = opacity;

}

else if (ct < num\*2)

{

opacity = this.Opacity + 0.1;

this.Opacity = opacity;

}

else if( ct == num\*2)

{

ct = 0;

}

}

}

}

Problem 3

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace Problem3

{

public class CNode

{

public int X, Y;

}

public partial class Form1 : Form

{

List<CNode> LTop = new List<CNode>();

List<CNode> LBottom = new List<CNode>();

int flag = 0;

int pos = 0;

int up = 0; //up not clicked

int top = 0;

int bottom = 0;

public Form1()

{

this.BackColor = Color.FromArgb(255, 255, 255);

this.MouseDown += MyMouseDown;

}

private void MyMouseDown(object sender, MouseEventArgs e)

{

if (e.Button == MouseButtons.Left)

{

if (flag == 0)

{

pos = e.Y;

this.Text = pos.ToString();

flag = 1;

}

else

{

if(up == 0 && e.Y < pos)

{

CNode pnn = new CNode();

pnn.X = e.X;

pnn.Y = e.Y;

LTop.Add(pnn);

up = 1; //up clicked

}

else if(up == 1 && e.Y > pos)

{

CNode pnn = new CNode();

pnn.X = e.X;

pnn.Y = e.Y;

LBottom.Add(pnn);

up = 0; //remove up click to make it not clicked

}

else

{

MessageBox.Show("Error");

}

}

}

else

{

if(top == 0)

{

MessageBox.Show("Top Points: ");

}

while (top < LTop.Count)

{

MessageBox.Show($"{LTop[top].X}, {LTop[top].Y}");

top++;

}

if(bottom == 0)

{

MessageBox.Show("Bottom Points: ");

}

while (bottom < LBottom.Count)

{

MessageBox.Show($"{LBottom[bottom].X}, {LBottom[bottom].Y}");

bottom++;

}

}

}

}

}