Problem 7

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 Problem7

{

public partial class Form1 : Form

{

public class CNode

{

public int R, G, B;

}

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

int ctClicks = 0;

int posi = 0;

CNode pnn;

public Form1()

{

this.MouseDown += Form1\_MouseDown;

this.KeyDown += Form1\_KeyDown;

}

private void Form1\_MouseDown(object sender, MouseEventArgs e)

{

if (e.Button == MouseButtons.Left)

{

if (e.X < 255)

{

ctClicks++;

if (ctClicks == 1)

{

pnn = new CNode();

pnn.R = e.X;

}

if (ctClicks == 2)

{

pnn.G = e.X;

}

if (ctClicks == 3)

{

pnn.B = e.X;

this.Text = pnn.R.ToString() + "," + pnn.G.ToString() + "," + pnn.B.ToString();

LColor.Add(pnn);

ctClicks = 0;

}

}

}

}

private void Form1\_KeyDown(object sender, KeyEventArgs e)

{

this.BackColor = Color.FromArgb(LColor[posi].R, LColor[posi].G, LColor[posi].B);

switch (e.KeyCode)

{

case Keys.Up:

posi++;

if(posi == LColor.Count)

{

posi = 0;

}

break;

case Keys.Down:

posi--;

if (posi < 0)

{

posi = LColor.Count - 1;

}

break;

}

}

}

}

Problem 8

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 Problem7

{

public partial class Form1 : Form

{

public Form1()

{

this.KeyDown += Form1\_KeyDown;

}

int ctEnter = 0;

int f1 = 0, f2 = 0;

int forwards = 0;

int posx0 = 0, posx2 = 0, posy2 = 0;

List<Form1> L = new List<Form1>();

private void Form1\_KeyDown(object sender, KeyEventArgs e)

{

switch (e.KeyCode)

{

case Keys.Enter:

ctEnter++;

if(ctEnter == 1)

{

this.Location = new Point(600,270);

}

if (ctEnter == 2)

{

//up left 0

Form1 pnn = new Form1();

pnn.Show();

pnn.BackColor = Color.Orange;

pnn.Size = new Size(150, 150);

pnn.Location = new Point(this.Location.X, this.Location.Y - 150);

forwards = 0;

posx0 = this.Location.X;

pnn.Opacity = 0.7;

L.Add(pnn);

//up right 1

pnn = new Form1();

pnn.Show();

pnn.BackColor = Color.Green;

pnn.Size = new Size(150, 150);

pnn.Location = new Point(this.Location.X + 150, this.Location.Y - 150);

forwards = 0;

pnn.Opacity = 0.7;

L.Add(pnn);

//right up 2

pnn = new Form1();

pnn.Show();

pnn.BackColor = Color.Yellow;

pnn.Size = new Size(150, 150);

pnn.Location = new Point(this.Location.X + this.ClientSize.Width + 10, this.Location.Y);

forwards = 0;

posx2 = this.Location.X + this.ClientSize.Width + 10;

posy2 = this.Location.Y;

pnn.Opacity = 0.7;

L.Add(pnn);

//right down 3

pnn = new Form1();

pnn.Show();

pnn.BackColor = Color.Maroon;

pnn.Size = new Size(150, 150);

pnn.Location = new Point(this.Location.X + this.ClientSize.Width + 150, this.Location.Y + 140);

forwards = 0;

pnn.Opacity = 0.7;

L.Add(pnn);

//down right 4

pnn = new Form1();

pnn.Show();

pnn.BackColor = Color.Maroon;

pnn.Size = new Size(150, 150);

pnn.Location = new Point(this.Location.X + 150, this.Location.Y + this.ClientSize.Height + 40);

forwards = 0;

pnn.Opacity = 0.7;

L.Add(pnn);

//down left 5

pnn = new Form1();

pnn.Show();

pnn.BackColor = Color.Yellow;

pnn.Size = new Size(150, 150);

pnn.Location = new Point(this.Location.X, this.Location.Y + this.ClientSize.Height + 40);

forwards = 0;

pnn.Opacity = 0.7;

L.Add(pnn);

//left down 6

pnn = new Form1();

pnn.Show();

pnn.BackColor = Color.Green;

pnn.Size = new Size(150, 150);

pnn.Location = new Point(this.Location.X - 285, this.Location.Y + 140);

forwards = 0;

pnn.Opacity = 0.7;

L.Add(pnn);

//left up 7

pnn = new Form1();

pnn.Show();

pnn.BackColor = Color.Orange;

pnn.Size = new Size(150, 150);

pnn.Location = new Point(this.Location.X - 145, this.Location.Y);

forwards = 0;

pnn.Opacity = 0.7;

L.Add(pnn);

}

break;

case Keys.D1:

f1 = 1;

f2 = 0;

break;

case Keys.D2:

f2 = 1;

f1 = 0;

break;

case Keys.Space:

if (f1 == 1)

{

if (forwards == 0)

{

L[2].Location = new Point(L[2].Location.X + 1, L[2].Location.Y + 1);

L[3].Location = new Point(L[3].Location.X - 1, L[3].Location.Y - 1);

L[6].Location = new Point(L[6].Location.X + 1, L[6].Location.Y - 1);

L[7].Location = new Point(L[7].Location.X - 1, L[7].Location.Y + 1);

}

if (forwards == 1)

{

L[2].Location = new Point(L[2].Location.X - 1, L[2].Location.Y - 1);

L[3].Location = new Point(L[3].Location.X + 1, L[3].Location.Y + 1);

L[6].Location = new Point(L[6].Location.X - 1, L[6].Location.Y + 1);

L[7].Location = new Point(L[7].Location.X + 1, L[7].Location.Y - 1);

}

if (L[3].Location.X == posx2 && L[3].Location.Y == posy2)

{

forwards = 1;

}

if (L[2].Location.X == posx2 && L[2].Location.Y == posy2)

{

forwards = 0;

}

}

if (f2 == 1)

{

if (forwards == 0 )

{

L[0].Location = new Point(L[0].Location.X + 1, L[0].Location.Y);

L[1].Location = new Point(L[1].Location.X - 1, L[1].Location.Y);

L[4].Location = new Point(L[4].Location.X - 1, L[4].Location.Y);

L[5].Location = new Point(L[5].Location.X + 1, L[5].Location.Y);

}

if (forwards == 1 )

{

L[0].Location = new Point(L[0].Location.X - 1, L[0].Location.Y);

L[1].Location = new Point(L[1].Location.X + 1, L[1].Location.Y);

L[4].Location = new Point(L[4].Location.X + 1, L[4].Location.Y);

L[5].Location = new Point(L[5].Location.X - 1, L[5].Location.Y);

}

if (L[1].Location.X == posx0)

{

forwards = 1;

}

if (L[0].Location.X == posx0)

{

forwards = 0;

}

}

break;

}

}

}

}

Problem 9

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 Problem9

{

public partial class Form1 : Form

{

public Form1()

{

this.MouseDown += Form1\_MouseDown;

this.BackColor = Color.White;

}

int x1 = 0, y1 = 0, x2 = 0, y2 = 0;

int ctClicks = 0;

void DrawEllipse(int x, int y, int width, int height, int ctClicks)

{

Graphics g = this.CreateGraphics();

if (ctClicks == 1)

{

g.Clear(Color.White);

}

SolidBrush brush = new SolidBrush(Color.Blue);

g.FillEllipse(brush, x, y, width, height);

}

void DrawShape(int firstX, int firsty, int w, int h)

{

Graphics g = this.CreateGraphics();

Pen P = new Pen(Color.Green);

g.DrawRectangle(P, firstX, firsty, w, h);

SolidBrush brush = new SolidBrush(Color.Green);

g.FillRectangle(brush, firstX, firsty, w, h);

}

private void Form1\_MouseDown(object sender, MouseEventArgs e)

{

ctClicks++;

if (ctClicks == 1)

{

x1 = e.X;

y1 = e.Y;

DrawEllipse(x1, y1, 10, 10, 1);

}

if (ctClicks == 2)

{

x2 = e.X;

y2 = e.Y;

int dx = x2 - x1;

int dy = y2 - y1;

if (dx < 0 && dy > 0)

{

dx = x1 - x2;

DrawShape(x2, y1, dx, dy);

}

else if (dx < 0 && dy < 0)

{

dx = x1 - x2;

dy = y1 - y2;

DrawShape(x2, y2, dx, dy);

}

else if (dx > 0 && dy < 0)

{

dy = y1 - y2;

DrawShape(x1, y2, dx, dy);

}

else if (dx > 0 && dy > 0)

{

DrawShape(x1, y1, dx, dy);

}

DrawEllipse(x2, y2, 10, 10, 2);

ctClicks = 0;

}

}

}

}