Lucrare laborator L3 + schema logica refacuta de la lucrarea L2

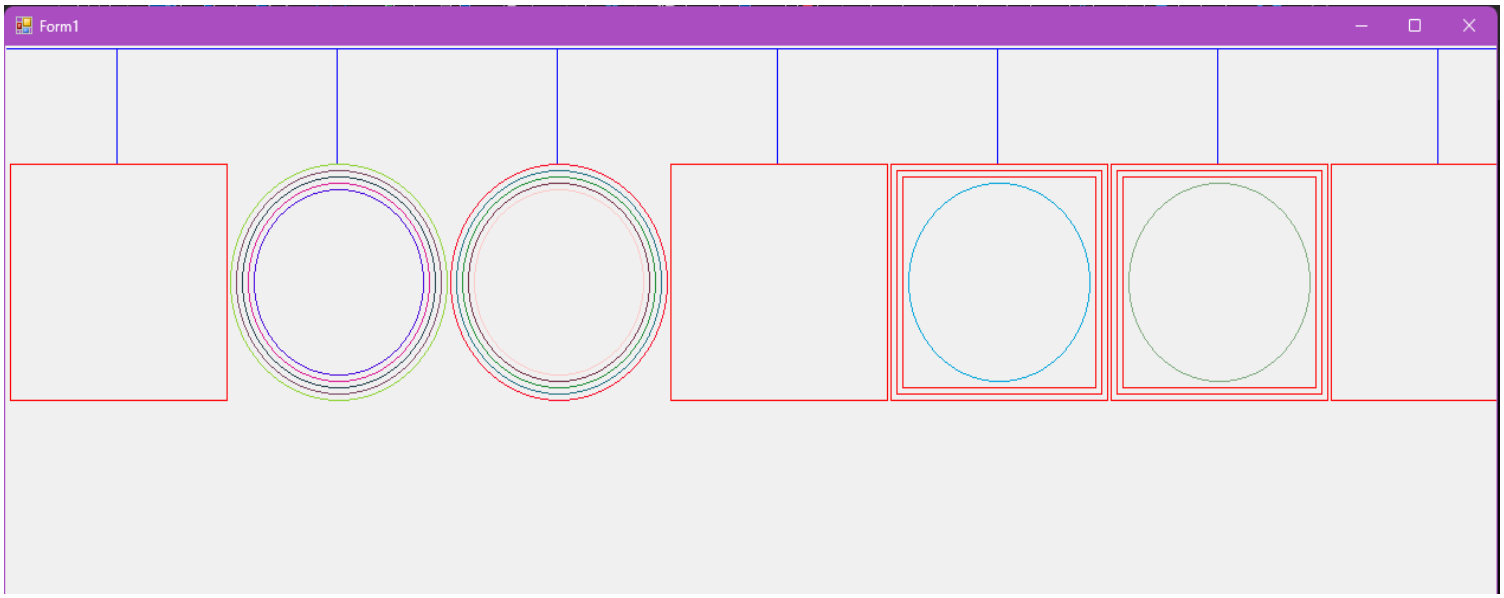
Codul in C++ folosind doua structure repetitive imbricate:

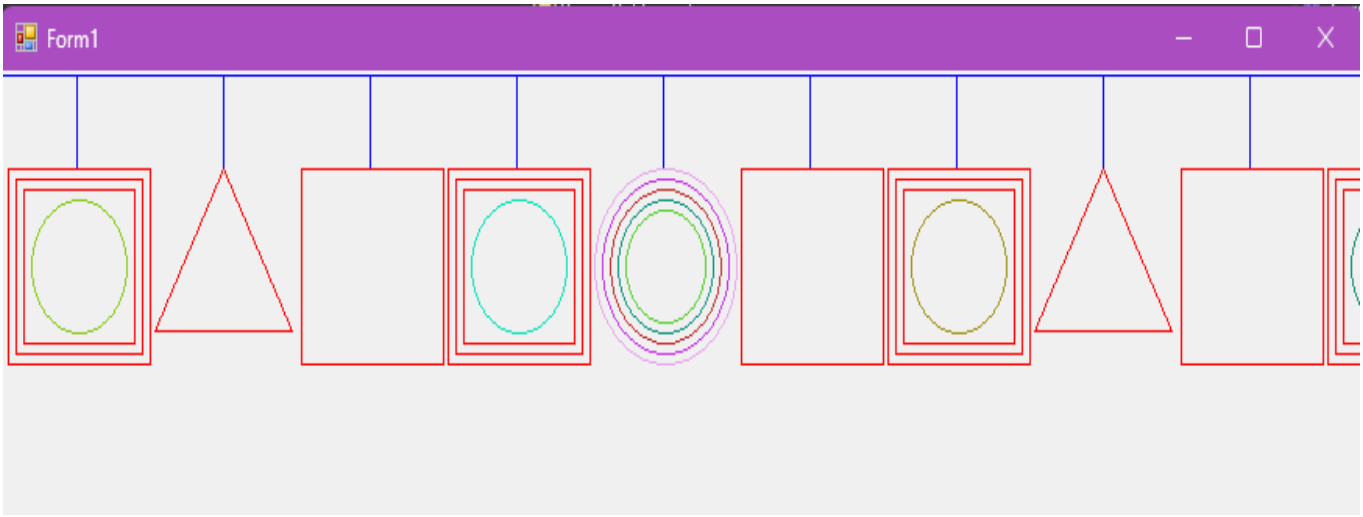
```cpp
#include <iostream>
using namespace std;
int main()
{
    int i,j,x=-1,n;
    cin>>n;
    for(i=1;i<=n;i++)
    {
        for(j=1;j<n;j++)
        {
            if(j>=n/2-x && j<=n/2+x)
                cout<<' ';
            else
            {
                if(i<=n/2)
                {
                    if(j<n/2)
                        cout<<"\033[32m"<<'*';
                    else cout<<"\033[35m"<<'*';
                }
                else
                {
                    if(j<n/2)
                        cout<<"\033[33m"<<'*';
                    else cout<<"\033[34m"<<'*';
                }
            }
        }
        if(i<=n/2)
            x++;
        else x--;
        cout<<endl;
    }
    return 0;
}
```

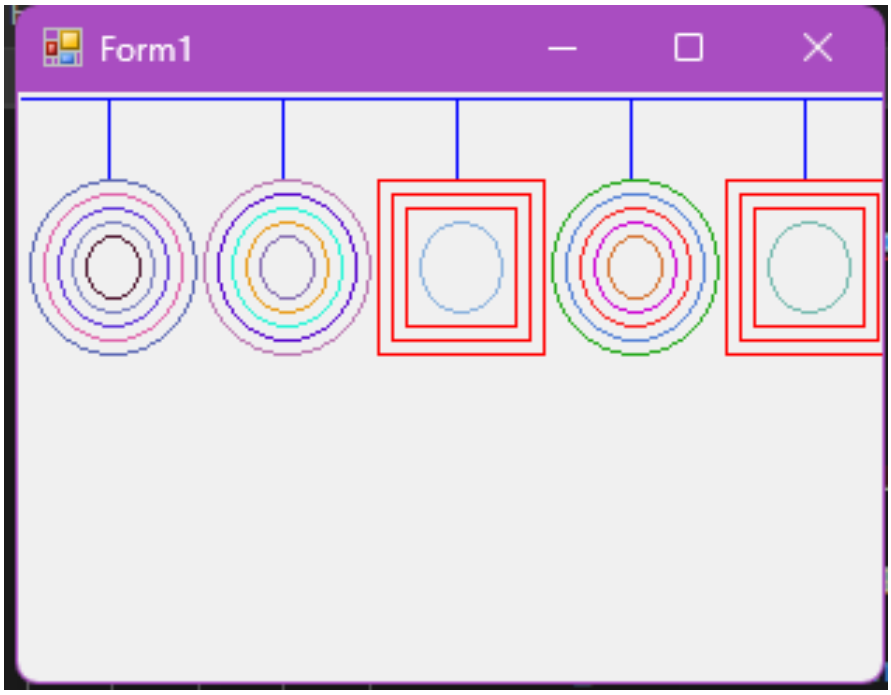
Caseta de afisare:

Codul C# ce afiseaza un ornament rescalabil pentru sarbatorile de iarna:

```
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 ProiectL3
{
    3 references
    public partial class Form1 : Form
    {
        1 reference
        public Form1()
        {
            InitializeComponent();
        }

        1 reference
        private void Form1_Paint(object sender, PaintEventArgs e)
        {
            System.Drawing.Graphics desen;
            System.Random n;
            System.Drawing.Pen creion_multicolor, creion_verde, creion_indigo, creion_rosu;
            System.Drawing.SolidBrush radiera;
            desen = this.CreateGraphics();
            n = new System.Random();
            creion_verde = new System.Drawing.Pen(System.Drawing.Color.SpringGreen);
            creion_rosu = new System.Drawing.Pen(System.Drawing.Color.Red);
            creion_indigo = new System.Drawing.Pen(System.Drawing.Color.Indigo);
            radiera = new System.Drawing.SolidBrush(this.BackColor);
            desen.Clear(BackColor);
            int w, h, x0, y0;
            w = Width / 3;
            h = Height / 4;
            x0 = 1;
            while (x0 <= Width)
            {
                int nr = n.Next(5);
                desen.DrawLine(creion_multicolor, x0, 2, x0 + h, 2);
                desen.DrawLine(creion_verde, (x0 + h) - h / 2, 2, (x0 + h) - h / 2, h / 2);
                switch (nr)
                {
                    case 1:
                        for (int i = 0; i < 15; i += 5)
                        {
                            desen.DrawRectangle(creion_rosu, x0 + 3 + i, h / 2 + i, h - 3 - 2 * i, h - 2 * i);
                        }
                        creion_multicolor = new
                        System.Drawing.Pen(Color.FromArgb(n.Next(255), n.Next(255), n.Next(255)));
                        desen.DrawEllipse(creion_multicolor, x0 + 18, h / 2 + 15, h - 33, h - 30);
                        break;
                    case 2:
                        desen.DrawLine(creion_rosu, x0 + 3, h * 4 / 3, x0 + h / 2, h / 2);
                        desen.DrawLine(creion_verde, x0 + 3, h * 4 / 3, x0 + h - 3, h * 4 / 3);
                        desen.DrawLine(creion_verde, x0 + h / 2, h / 2, x0 + h - 3, h * 4 / 3);
                        break;
                    case 3:
                        for (int i = 0; i < 25; i += 5)
                        {
                            creion_multicolor = new
                            System.Drawing.Pen(Color.FromArgb(n.Next(255), n.Next(255), n.Next(255)));
                            desen.DrawEllipse(creion_multicolor, x0 + 3 + i, h / 2 + i, h - 3 - i * 2, h - i * 2);
                        }
                        break;
                        desen.DrawLine(creion_rosu, x0 + h / 2, h * 11 / 7, x0 + h - 3, h);
                        desen.DrawLine(creion_verde, x0 + 3, h, x0 + h / 2, h * 11 / 7);
                        desen.DrawLine(creion_indigo, x0 + h / 2, h / 2, x0 + h - 3, h);
                        break;
                    default:
                        desen.DrawRectangle(creion_multicolor, x0 + 3, h / 2, h - 3, h);
                        break;



                }
                x0 += h;

                }
            }
        }
}
```


Form1


Form1


Form1

Schema logica refacuta pentru lucrarea L2: