

PROIECT PROGRAMARE [100%]

„Joc de lumini”

Al cincilea proiect-programare , reprezentand 100%

Ujica Septimiu

Descriere:

Program realizat in windows forms application , realizeaza un joc de lumini aleatoare pe marginile form-ului. Acestea sunt controlate de un buton de pornire , respectiv oprire.

Pentru a face aceste lucruri posibile am folosit intai un timer. In el a fost introdus codul principal.

Jocul de lumini este efectiv un dreptunghi in jurul form-ului nostru format din cercuri , fiecare cerc avand o culoare diferita , schimbandu-se in functie de valoarea timer-ului.

Pentru a crea acest dreptunghi de cercuri intai am declarat patru variabile , pentru fiecare latura a dreptunghiului , fiecare fiind creata separat cu cate o instructiune „for”.

Intai am creat spatiul pentru desenat intitulat „desen” , apoi am creat creionul „pix”. Urmatorul lucru ce trebuie facut este sa scriem sintaxa pentru a crea cercurile colorate random , la intamplare. Pentru asta am folosit : `System::Random^ x = gcnew System::Random();` .

Intram in for , avand nevoie de patru. In fiecare este un cod asemanator , doar coordonatele fiind diferite. Pentru a crea fiecare cerc colorat diferit , in interiorul fiecarui for am introdus elementul ce randomizeaza culorile si anume :

`System::Drawing::Color::FromArgb(x->Next(256),x->Next(256),x->Next(256))` , deasemenea dand pixului o grosime mai mare pentru a fi intreg , dand aceasta simulare de joc de lumini , oarecum. In ultimul pas trebuie doar sa desenam elipsa , iar la coordonate sa le dam fixe toate , cu exceptia latimii care este flexibila si variaza in functie de expresia de la for , in cazul nostru: `a=5; a<=this->Width-10; a+=40.`

Pentru a nu se desena cercurile chiar de la capat , am dat prima valoare 5 a lui a pentru a arata mai estetic , la fel si pentru celalalt capat , punand width-10. Aceeasi operatie se repeta de inca trei ori sa

creem toate laturile singura diferenta fiind ca la al doilea for schimbam inaltimea cu latimea la expresia for-ului , iar la ultimele doua , deoarece sunt jos , respectiv in dreapta trebuie sa mai declaram inca doua variabile pentru a crea si cealalta coordonata in cazul nostru : `int q=this->Width-40;` , respectiv la cealalta schimbam latimea cu inaltimea in cod , rezultand un dreptunghi multicolor ce isi schimba culorile regulat.

La sfarsitul codului stergem desenul si pix-ul create pentru a elibera memoria.

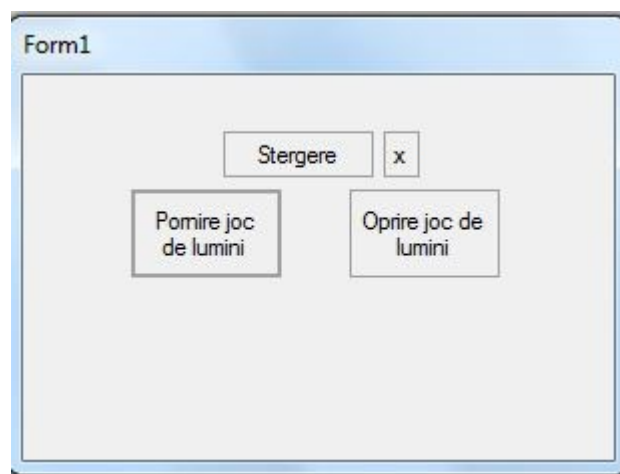
Pasul urmator e de a crea 3 butoane. Unul pentru a activa acest joc de lumini , unul pentru dezactivare , iar celalalt pentru a sterge ecranul.

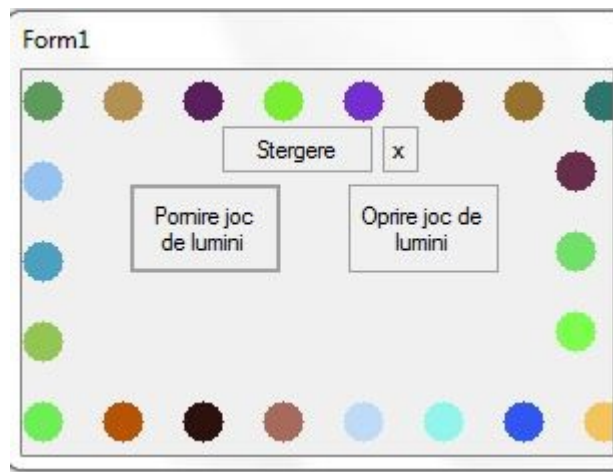
Am mai adaugat inca un buton optional de inchidere a programului , nefiind obligatorie prezenta lui.

La primul buton am dat enable la timer , pornind astfel crearea dreptunghiului de cercuri si schimbarea culorilor , la al doilea buton are loc oprirea culorilor ramanand doar dreptunghiul desenat neschimandu-si culorile deloc , iar al treilea sterge desenul de tot ramanand programul gol.

Deasemenea pentru a nu incarca memoria , lucru ce nu ne dorim , am mai pus inca un timer ce este activ cat timp se deseneaza dreptunghiul ce sterge din cand in cand desenul pentru a nu incarca memoria prea tare , evitand astfel o blocare de memorie , sau blocarea programului.

[Poze:](#)





Cod sursa:

#pragma once

```
namespace School_Project5 {

    using namespace System;
    using namespace System::ComponentModel;
    using namespace System::Collections;
    using namespace System::Windows::Forms;
    using namespace System::Data;
    using namespace System::Drawing;

    /// <summary>
    /// Summary for Form1
    /// </summary>
    public ref class Form1 : public System::Windows::Forms::Form
    {
    public:
        Form1(void)
        {
            InitializeComponent();
            //
            //TODO: Add the constructor code here
            //
        }

    protected:
        /// <summary>
        /// Clean up any resources being used.
        /// </summary>
        ~Form1()
        {
            if (components)
            {
                delete components;
            }
        }

    private: System::Windows::Forms::Timer^ timer1;
    private: System::Windows::Forms::Button^ button1;
    private: System::Windows::Forms::Button^ button2;
    private: System::Windows::Forms::Timer^ timer2;
    private: System::Windows::Forms::Button^ button3;
    private: System::Windows::Forms::Button^ button4;
    }
```

```

protected:
private: System::ComponentModel::IContainer^ components;

protected:

private:
    /// <summary>
    /// Required designer variable.
    /// </summary>

#pragma region Windows Form Designer generated code
    /// <summary>
    /// Required method for Designer support - do not modify
    /// the contents of this method with the code editor.
    /// </summary>
    void InitializeComponent(void)
    {
        this->components = (gcnew System::ComponentModel::Container());
        this->timer1 = (gcnew System::Windows::Forms::Timer(this->components));
        this->button1 = (gcnew System::Windows::Forms::Button());
        this->button2 = (gcnew System::Windows::Forms::Button());
        this->timer2 = (gcnew System::Windows::Forms::Timer(this->components));
        this->button3 = (gcnew System::Windows::Forms::Button());
        this->button4 = (gcnew System::Windows::Forms::Button());
        this->SuspendLayout();
        //
        // timer1
        //
        this->timer1->Interval = 200;
        this->timer1->Tick += gcnew System::EventHandler(this,
&Form1::timer1_Tick);
        //
        // button1
        //
        this->button1->FlatStyle = System::Windows::Forms::FlatStyle::Popup;
        this->button1->Location = System::Drawing::Point(54, 57);
        this->button1->Name = L"button1";
        this->button1->Size = System::Drawing::Size(75, 44);
        this->button1->TabIndex = 0;
        this->button1->Text = L"Pornire joc de lumini";
        this->button1->UseVisualStyleBackColor = true;
        this->button1->Click += gcnew System::EventHandler(this,
&Form1::button1_Click);
        //
        // button2
        //
        this->button2->Anchor =
static_cast<System::Windows::Forms::AnchorStyles>((System::Windows::Forms::AnchorStyles::Top
| System::Windows::Forms::AnchorStyles::Right));
        this->button2->FlatStyle = System::Windows::Forms::FlatStyle::Popup;
        this->button2->Location = System::Drawing::Point(163, 57);
        this->button2->Name = L"button2";
        this->button2->Size = System::Drawing::Size(75, 44);
        this->button2->TabIndex = 2;
        this->button2->Text = L"Oprire joc de lumini";
        this->button2->UseVisualStyleBackColor = true;
        this->button2->Click += gcnew System::EventHandler(this,
&Form1::button2_Click);
        //
        // timer2
        //
        this->timer2->Interval = 1500;
        this->timer2->Tick += gcnew System::EventHandler(this,

```

```

&Form1::timer2_Tick);
    //
    // button3
    //
    this->button3->Anchor =
static_cast<System::Windows::Forms::AnchorStyles>(((System::Windows::Forms::AnchorStyles::To
p | System::Windows::Forms::AnchorStyles::Left)
    | System::Windows::Forms::AnchorStyles::Right));
    this->button3->FlatStyle = System::Windows::Forms::FlatStyle::Popup;
    this->button3->Location = System::Drawing::Point(100, 28);
    this->button3->Name = L"button3";
    this->button3->Size = System::Drawing::Size(75, 23);
    this->button3->TabIndex = 3;
    this->button3->Text = L"Stergere";
    this->button3->UseVisualStyleBackColor = true;
    this->button3->Click += gcnew System::EventHandler(this,
&Form1::button3_Click);
    //
    // button4
    //
    this->button4->FlatStyle = System::Windows::Forms::FlatStyle::Popup;
    this->button4->Location = System::Drawing::Point(180, 28);
    this->button4->Name = L"button4";
    this->button4->Size = System::Drawing::Size(18, 23);
    this->button4->TabIndex = 4;
    this->button4->Text = L"x";
    this->button4->UseVisualStyleBackColor = true;
    this->button4->Click += gcnew System::EventHandler(this,
&Form1::button4_Click);
    //
    // Form1
    //
    this->AutoScaleDimensions = System::Drawing::SizeF(6, 13);
    this->AutoScaleMode = System::Windows::Forms::AutoScaleMode::Font;
    this->BackColor = System::Drawing::SystemColors::Control;
    this->ClientSize = System::Drawing::Size(295, 192);
    this->ControlBox = false;
    this->Controls->Add(this->button4);
    this->Controls->Add(this->button3);
    this->Controls->Add(this->button2);
    this->Controls->Add(this->button1);
    this->MinimumSize = System::Drawing::Size(300, 208);
    this->Name = L"Form1";
    this->Text = L"Form1";
    this->ResizeEnd += gcnew System::EventHandler(this,
&Form1::Form1_ResizeEnd);
    this->ResumeLayout(false);

}
#pragma endregion
private: System::Void timer1_Tick(System::Object^ sender, System::EventArgs^ e) {
    int a , b , c , d;
    System::Drawing::Graphics^ desen;
    desen = this->CreateGraphics();
    System::Drawing::Pen^ pix;
    System::Random^ x = gcnew System::Random();
    for(a=5; a<=this->Width-10; a+=40)
    {
        pix=gcnew
System::Drawing::Pen(System::Drawing::Color::FromArgb(x->Next(256),x->Next(256),x-
>Next(256)));
        pix->Width=10;
        desen->DrawEllipse(pix,a,10,10,10);
    }
    for(b=10; b<=this->Height-10; b+=40)

```

```

        {
            pix=gcnew
System::Drawing::Pen(System::Drawing::Color::FromArgb(x->Next(256),x->Next(256),x-
>Next(256)));
            pix->Width=10;
            desen->DrawEllipse(pix,5,b,10,10);
        }
        for(c=45; c<=this->Width-10; c+=40)
        {
            pix=gcnew
System::Drawing::Pen(System::Drawing::Color::FromArgb(x->Next(256),x->Next(256),x-
>Next(256)));
            pix->Width=10;
            int s=this->Height-60;
            desen->DrawEllipse(pix,c,s,10,10);
        }
        for(d=45; d<=this->Height-90; d+=40)
        {
            pix=gcnew
System::Drawing::Pen(System::Drawing::Color::FromArgb(x->Next(256),x->Next(256),x-
>Next(256)));
            pix->Width=10;
            int q=this->Width-40;
            desen->DrawEllipse(pix,q,d,10,10);
        }
        delete pix;
        delete desen;
    }
    private: System::Void button1_Click(System::Object^ sender, System::EventArgs^ e) {
        this->timer1->Enabled=true;
        this->timer2->Enabled=true;
    }
    private: System::Void button2_Click(System::Object^ sender, System::EventArgs^ e) {
        this->timer1->Enabled=false;
        this->timer2->Enabled=false;
    }
    private: System::Void Form1_ResizeEnd(System::Object^ sender, System::EventArgs^ e) {
        System::Drawing::Graphics^ desen;
        desen = this->CreateGraphics();
        desen->Clear(System::Drawing::Color(this->BackColor));
    }
    private: System::Void timer2_Tick(System::Object^ sender, System::EventArgs^ e) {
        System::Drawing::Graphics^ desen;
        desen = this->CreateGraphics();
        desen->Clear(System::Drawing::Color(this->BackColor));
    }
    private: System::Void button3_Click(System::Object^ sender, System::EventArgs^ e) {
        System::Drawing::Graphics^ desen;
        desen = this->CreateGraphics();
        desen->Clear(System::Drawing::Color(this->BackColor));
    }
    private: System::Void button4_Click(System::Object^ sender, System::EventArgs^ e) {
        this->Close();
    }
};
}

```