**COLEGIUL NAȚIONAL “VASILE ALECSANDRI” GALAȚI**

Proiect pentru susținerea atestatului profesional la informatică

Joc – Profesorul PAC-MAN

**Filieră**: teoretică

**Profil**: real

**Specializare**: matematică-informatică

Coordonator: Autor:

Stan Mihaela Bubuianu Adrian-Florin

Clasa a XII-a E

Cuprins

[1. TEMA, UTILITATEA APLICAȚIEI ȘI MOTIVAȚIA ALEGERII LUCRĂRII 3](#_Toc132666472)

[2. PREZENTAREA TEHNOLOGIILOR INFORMATICE UTILIZATE 3](#_Toc132666473)

[3. PREZENTAREA TEHNOLOGIILOR INFORMATICE UTILIZATE 4](#_Toc132666474)

[4. INTERFAȚA UTILIZATOR ȘI DESIGN-UL APLICAȚIEI 5](#_Toc132666475)

[5. STRUCTURA PRODUSULUI INFORMATIC 5](#_Toc132666476)

[1. Prima fereastră – pagina de prezentare 6](#_Toc132666477)

[2. A doua fereastră - rezolvarea celor 20 de grile 6](#_Toc132666478)

[3. A treia fereastră - jocul PAC-MAN 7](#_Toc132666479)

[6. BIBLIOGRAFIE 8](#_Toc132666480)

[ANEXE 9](#_Toc132666481)

# TEMA, UTILITATEA APLICAȚIEI ȘI MOTIVAȚIA ALEGERII LUCRĂRII

Să se realizeze o aplicație de tip Windows Form, folosind mediul de programare Visual C# 2022.

Aplicația dezvoltată va reprezenta un soft educativ, dar și de divertisment, prin care utilizatorul își va testa cunoștințele la informatică prin intermediul a 20 de grile, având oricând ocazia să ia o pauză, jucându-se PAC-MAN.

Am ales acest proiect nu numai pentru a oferi un suport educativ elevilor de liceu care doresc să își afle nivelul la informatică, dar și pentru a atrage atenția asupra importanței de a lua mici pauze în timpul oricărei perioade de studiu.

# PREZENTAREA TEHNOLOGIILOR INFORMATICE UTILIZATE

Am folosit limbajul de programare C#, acesta fiind asemănator cu cel studiat în anii precedenti (C++), în mediul Microsoft Visual Studio 2022.

**Limbajul C#** a fost dezvoltat de o echipă restrânsă de ingineri de la Microsoft, echipă din care s-a evidențiat Anders Hejlsberg (autorul limbajului Turbo Pascal și membru al echipei care a proiectat Borland Delphi). C# este un limbaj simplu, cu circa 80 de cuvinte cheie și 12 tipuri de date predefinite. El permite programarea structurată, modulară și orientată obiectual, conform conceptelor moderne ale programării profesioniste.

Principiile de bază ale programării pe obiecte (ÎNCAPSULARE, MOȘTENIRE, POLIMORFISM) sunt elemente fundamentale ale programării C#. În mare, limbajul moștenește sintaxa și principiile de programare din C++. Sunt o serie de tipuri noi de date sau funcțiuni diferite ale datelor din C++, iar în spiritul realizării unor secvente de cod sigure (safe), unele funcțiuni au fost adăugate (de exemplu, interfețe si delegări), diversificate (tipul struct), modificate (tipul string) sau chiar eliminate (moștenirea multiplă și pointerii către funcții). Unele funcțiuni (cum ar fi accesul direct la memorie folosind pointeri) au fost păstrate, dar secvențele de cod corespunzătoare se consideră ”nesigure”.

C# derivă din doua dintre cele mai de succes limbaje de programare: C și C++. De asemenea, limbajul este o “ruda” apropiată a limbajului Java.

În C# toate tipurile de date sunt de fapt clase derivate direct sau indirect din clasa System.Object. Limbajul permite utilizarea unor nume alternative pentru tipurile simple de date. Declararea şi iniţializarea variabilelor (pentru tipuri simple) se face la fel ca în C++.

Limbajul poate efectua conversii între tipurile de date numerice: automat în cazul în care tipul destinaţie este mai puternic decât tipul sursă sau explicit dacă există posibilitatea pierderii de informaţii (ex: convertire din long în int).

Visual Studio include un set complet de instrumente de dezvoltare pentru generarea de aplicații ASP.NET, Servicii Web XML, aplicații desktop și aplicații mobile. Visual Basic, Visual C++, Visual C# și Visual J# - toate folosesc același mediu de dezvoltare integrat (IDE) care le permite partajarea instrumentelor și facilitează crearea de soluții folosind mai multe limbaje de programare. Aceste limbaje permit să beneficieze de caracteristicile .NET Framework care oferă acces la tehnologii cheie care simplifica dezvoltarea de aplicații web ASP și XML Web Services cu Visual Web Developer.

**Visual Studio 2022**

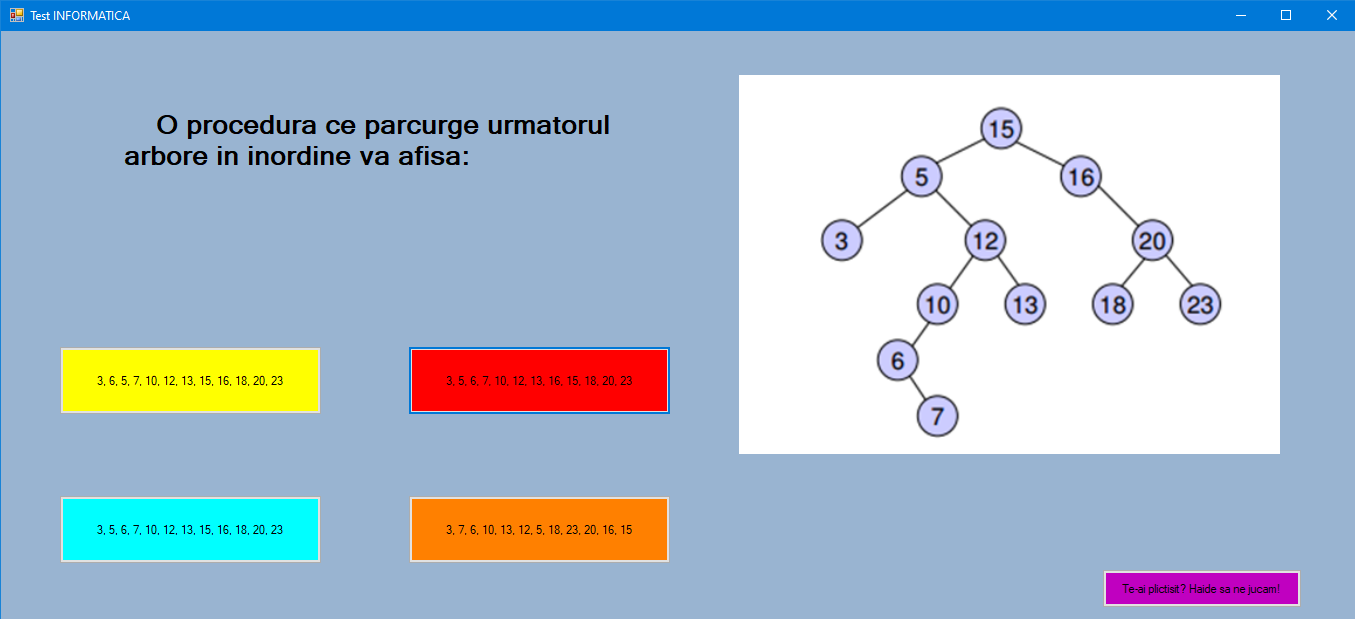
Pe 19 aprilie 2021, Microsoft a anunțat apariția Visual Studio 2022 (versiunea 17). Este prima versiune care rulează ca un proces pe 64 de biți, permițând procesului principal Visual Studio să acceseze mai mult de 4 GB de memorie, prevenind excepțiile de lipsă de memorie care ar putea apărea în cazul proiectelor mari. Această versiune este disponibilă spre a fi descarcată din data de 8 noiembrie 2021.

# PREZENTAREA TEHNOLOGIILOR INFORMATICE UTILIZATE

Programul funcționează pe orice calculator cu .NET Framework 4, dar este recomandat, totuși, să îndeplinească următoarele cerințe:

* Hardware
  + Procesor: 1.5 Ghz
  + Memorie RAM: 1 GB
* Software
  + Windows XP Service Pack 3, pe 32 sau 64 de biți

# INTERFAȚA UTILIZATOR ȘI DESIGN-UL APLICAȚIEI



- Exemplu fereastră -

Această aplicație conține formuri pentru afișare, butoane: pentru navigarea de la o pagină la alta, dar și pentru alegerea răspunsurilor la grilele propuse, închiderea ferestrelor realizându-se prin folosirea butonului X din colțul dreapta sus, iar a aplicației prin utilizarea butonului X din cel de-al doilea form (cel cu testarea cunostintelor la informatică).

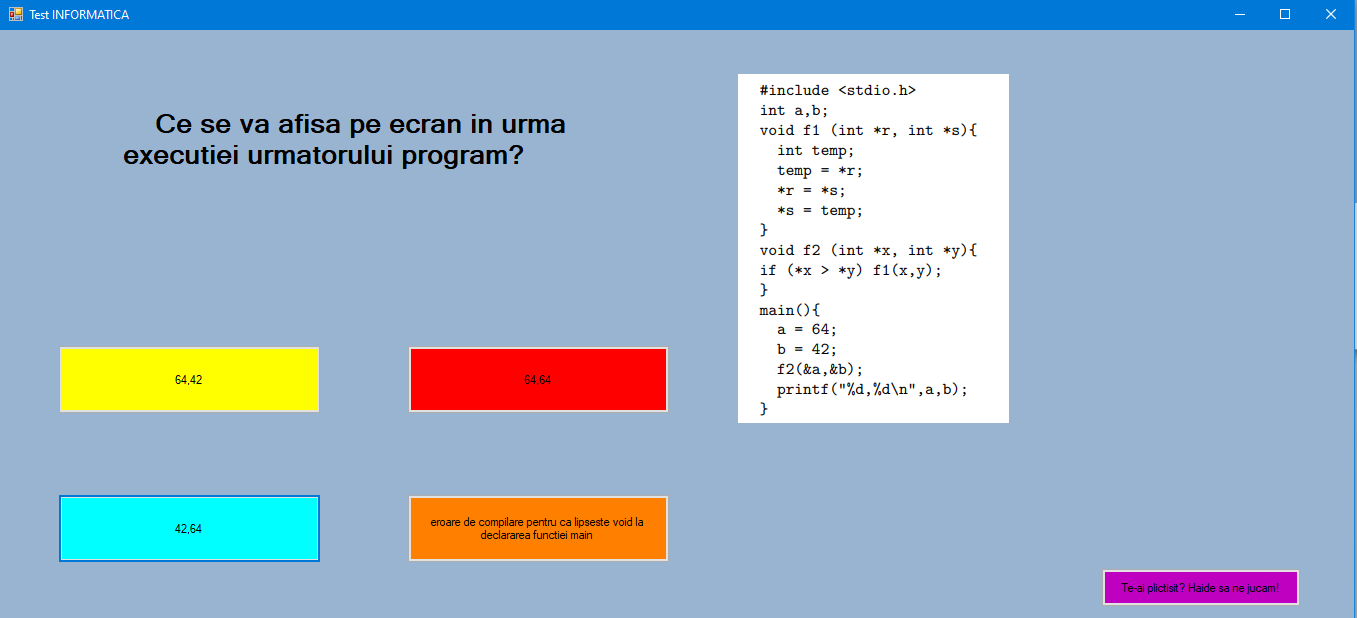
# STRUCTURA PRODUSULUI INFORMATIC

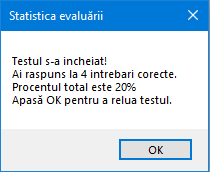
## Prima fereastră – pagina de prezentare



Prin apăsarea butonului – „Să începem!” se va accesa fereastra principală – cu grilele la informatică.

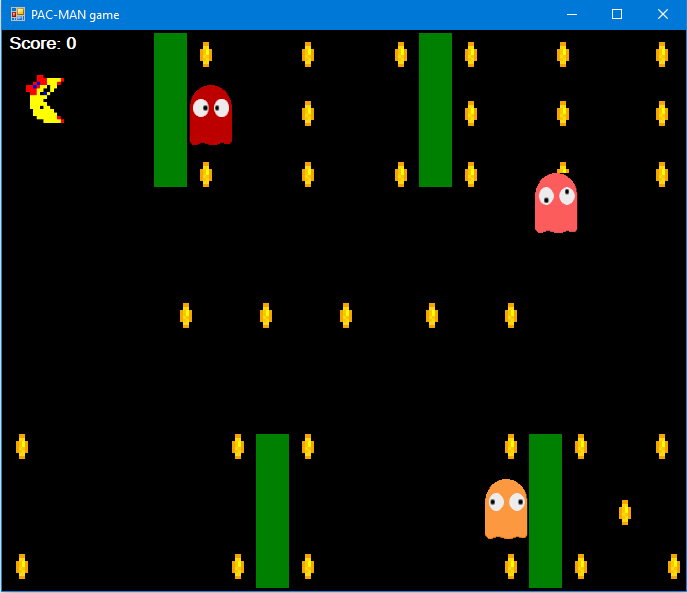
## 2. A doua fereastră - rezolvarea celor 20 de grile



 Această fereastră cuprinde 20 de grile, fiecare având câte 4 răspunsuri propuse, din care unul singur este corect. La sfârșitul rezolvării tuturor grilelor, se va afișa un mesaj ce conține o scurtă statistică a evaluării utilizatorului.

Pe măsura rezolvării grilelor, utilizatorul poate interactiona cu butonul – „Te-ai plictisit? Haide să ne jucăm!”, prin care se accesează ultima fereastră – jocul PAC-MAN. De notat este că progresul acestuia în cadrul grilelor este salvat (atunci când se întoarce din fereastra aferentă jocului PAC-MAN, utilizatorul va avea de rezolvat grila la care rămăsese înainte să acceseze butonul către joc).

## 3. A treia fereastră - jocul PAC-MAN



Odată accesată aceasta fereastră, jocul pornește. Scopul acestuia este de a colecta toți cei 34 de galbeni, fără a fi atins de cele 3 fantome aflate în mișcare și fara a lovi zidurile verzi. În acest caz, se va afișa la final mesajul: „Ai câștigat!”. Totuși, dacă are loc unul dintre evenimentele nefavorabile, anterior menționate, jocul se oprește cu mențiunea: „Ai pierdut!”. Pentru a reîncepe jocul, trebuie apăsată tasta – ENTER. Atunci când utilizatorul consideră de cuviință că e timpul să revină la rezolvarea grilelor de informatică, tot ce trebuie el să facă este să apese butonul roșu „X” din colțul dreapta-sus.

# BIBLIOGRAFIE:

* Liviu Negrescu, Lavinia Negrescu –**Limbajul C# pentru începători** , Editura Albastră, 2005
* Mariana Miloșescu-**Manual de informatică C++-**clasa a XI-a-Editura Didactică și Pedagogică,2007, București
* Ioan Lungu-**Baze de date Oracle**, limbajul SQL, editura ASE, 2005, Bucuresti
* Surse de documentare online:

<https://msdn.microsoft.com/en-us/library/67ef8sbd.aspx>

[www.info.mcip.ro](http://www.info.mcip.ro)

[www.wikipedia.ro](http://www.wikipedia.ro)

# ANEXE

Program.cs – asigură deschiderea corectă a aplicației

using System;

using System.Collections.Generic;

using System.Linq;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace Profesorul\_Pac\_Man

{

internal static class Program

{

/// <summary>

/// The main entry point for the application.

/// </summary>

[STAThread]

static void Main()

{

Application.EnableVisualStyles();

Application.SetCompatibleTextRenderingDefault(false);

Pagina\_prezentare fp = new Pagina\_prezentare();

fp.Show();

Application.Run();

}

}

}

Pagina\_prezentare.cs

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 Profesorul\_Pac\_Man

{

public partial class Pagina\_prezentare : Form

{

public Pagina\_prezentare()

{

InitializeComponent();

}

private void label1\_Click(object sender, EventArgs e)

{

}

private void pictureBox1\_Click(object sender, EventArgs e)

{

}

private void button1\_Click(object sender, EventArgs e)

{

Form1 f1 = new Form1();

f1.Show(); // butonul trimite către fereastra cu grilele la

// informatică

this.Close(); // și închide pagina de prezentare

}

}

}

Form1.cs – fereastra cu grilele la informatică

using System;

using System.CodeDom.Compiler;

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;

using static System.Windows.Forms.VisualStyles.VisualStyleElement.TaskbarClock;

namespace Profesorul\_Pac\_Man

{

public partial class Form1 : Form

{

// variabilele pentru quiz

int correctAnswer;

int questionNumber = 1;

int score;

int percentage;

int totalQuestions;

public Form1()

{

InitializeComponent();

askQuestion(questionNumber);

totalQuestions = 20;

}

// ce se întâmplă atunci când se alege o variantă de răspuns

private void checkAnswerEvent(object sender, EventArgs e)

{

var senderObject = (Button)sender;

int buttonTag = Convert.ToInt32(senderObject.Tag);

if(buttonTag == correctAnswer) // daca varianta de răspuns aleasă e

// cea corectă, crește scorul

{

score++;

}

// dacă am ajuns la finalul testului, afisăm mesajul cu statistica

// evaluării

if(questionNumber == totalQuestions)

{

percentage = (int)Math.Round( (double)(score \* 100) / totalQuestions ) ;

MessageBox.Show(

"Testul s-a încheiat!" + Environment.NewLine +

"Ai raspuns la " + score + " întrebari corecte." + Environment.NewLine +

"Procentul total este " + percentage + "%" + Environment.NewLine +

"Apasă OK pentru a relua testul.", "Statistica evaluării"

);

// după apăsarea butonului OK, testul reîncepe

score = 0;

questionNumber = 0;

}

questionNumber++;

askQuestion(questionNumber);

}

// toate cele 20 de grile, cu raspunsurile propuse si varianta corectă

// afișarea intrebarii curente se realizează cu ajutorul instrucțiunii switch, care primește numărul grilei la care a ajuns utilizatorul

private void askQuestion(int qnum)

{

switch(qnum)

{

case 1:

pictureBox1.Image = Properties.Resources.Question\_1;

lblQuestion.Text = " Ce se va afișa pe ecran în urma execuției următorului program?";

button1.Text = "42,64";

button2.Text = "64,42";

button3.Text = "64,64";

button4.Text = "eroare de compilare, pentru că lipsește void la declararea funcției main";

correctAnswer = 1;

break;

case 2:

pictureBox1.Image = Properties.Resources.Question\_2;

lblQuestion.Text = " Ce se va afișa pe ecran în urma execuției următorului program?";

button1.Text = "300 800 310 600 900";

button2.Text = "300 900 320 600 1000";

button3.Text = "300 800 310 600 1000";

button4.Text = "niciuna din variante nu este corectă";

correctAnswer = 3;

break;

case 3:

pictureBox1.Image = Properties.Resources.Question\_3;

lblQuestion.Text = " Ce se va afișa pe ecran în urma execuției următorului program?";

button1.Text = "edccbf";

button2.Text = "edcbaf";

button3.Text = "bcdefg";

button4.Text = "niciuna din variante nu este corectă";

correctAnswer = 1;

break;

case 4:

pictureBox1.Image = Properties.Resources.Question\_4;

lblQuestion.Text = " Ce se va afișa pe ecran în urma execuției următorului program?";

button1.Text = "15";

button2.Text = "nimic (programul nu se compilează)";

button3.Text = "0";

button4.Text = "36";

correctAnswer = 2;

break;

case 5:

pictureBox1.Image = Properties.Resources.Question\_5;

lblQuestion.Text = " Se consideră secvența următoare, în care valorile lui n și x se presupun cunoscute, v este un vector cu elementele (v[0],v[1],...,v[n-1]).\r\n\t Precizați care dintre urmatoarele afirmații sunt adevărate:";

button1.Text = "Pentru n = 5, x = 3 ¸si v= (5,6,2,7,1), se afișează ultimele patru elemente nemodificate ale vectorului: 6 2 7 1.";

button2.Text = "Pentru n = 5, x = 1 și v = (2,1,3,1,4), se afișează: 2 3 1 4;";

button3.Text = "Secvența conține erori de sintaxă";

button4.Text = "Algoritmul șterge din vector elementul cu valoarea x, prin mutarea cu o poziție mai la stânga a elementelor aflate dupa el.";

correctAnswer = 4;

break;

case 6:

pictureBox1.Image = Properties.Resources.Question\_6;

lblQuestion.Text = " Ce valori se vor găsi în fișierul numere.txt după execuția următorului program?";

button1.Text = "11 17 5 23 11";

button2.Text = "20 4 10 14 34";

button3.Text = "11\r\n17\r\n5";

button4.Text = "11\r\n17\r\n5\r\n23\r\n11";

correctAnswer = 3;

break;

case 7:

pictureBox1.Image = Properties.Resources.Question\_7;

lblQuestion.Text = " O procedură ce parcurge următorul arbore în inordine va afișa:";

button1.Text = "3, 5, 6, 7, 10, 12, 13, 15, 16, 18, 20, 23";

button2.Text = "3, 6, 5, 7, 10, 12, 13, 15, 16, 18, 20, 23";

button3.Text = "3, 5, 6, 7, 10, 12, 13, 16, 15, 18, 20, 23";

button4.Text = "3, 7, 6, 10, 13, 12, 5, 18, 23, 20, 16, 15";

correctAnswer = 1;

break;

case 8:

pictureBox1.Image = Properties.Resources.Question\_8;

lblQuestion.Text = " Ce metodă de sortare este descrisă în pseudocodul următor?";

button1.Text = "HeapSort";

button2.Text = "MergeSort";

button3.Text = "BubbleSort";

button4.Text = "SwapSort";

correctAnswer = 3;

break;

case 9:

pictureBox1.Image = Properties.Resources.Question\_9;

lblQuestion.Text = " Ce se va afișa pe ecran în urma execuției următorului program:";

button1.Text = "Eroare de compilare";

button2.Text = "2 1 1 1";

button3.Text = "2 5 2 0";

button4.Text = "2 5 2 1";

correctAnswer = 4;

break;

case 10:

pictureBox1.Image = Properties.Resources.Question\_10;

lblQuestion.Text = " Ce puteți afirma despre următorul program Java?";

button1.Text = "Programul nu este corect, deoarece asignarea lui x, conform obiectului b, este ilegală in Java;";

button2.Text = "Afișează:\r\na.x=2 a.x=200;";

button3.Text = "Afișează:\r\na.x=0 a.x=100;";

button4.Text = "Afișează:\r\na.x=1 a.x=100;";

correctAnswer = 2;

break;

case 11:

pictureBox1.Image = Properties.Resources.Question\_11;

lblQuestion.Text = " Ce puteti spune despre următorul program Java?";

button1.Text = "Programul este corect și va afișa la execuție a=4, b=3;";

button2.Text = "Va apărea eroare la compilare, deoarece clasa C2 nu poate fi definită în metoda f() din clasa C1;";

button3.Text = "Va apărea eroare la compilare, deoarece în metoda g() nu putem accesa variabila locala a din metoda f();";

button4.Text = "Va apărea eroare la compilare, deoarece nu se creeaza în clasa Test un obiect de tip C1.C2";

correctAnswer = 3;

break;

case 12:

pictureBox1.Image = Properties.Resources.Question\_12;

lblQuestion.Text = " Care este ordinea de parcurgere a nodurilor grafului din figura 1, dacă se folosește metoda de vizitare în lățime (BFS) pornind din nodul 5?";

button1.Text = "5, 2, 1, 3, 4, 7, 6, 8";

button2.Text = "5, 2, 7, 1, 6, 3, 4, 8";

button3.Text = "5, 2, 1, 3, 5, 7, 6, 8";

button4.Text = "5, 7, 2, 6, 1, 8, 4, 3";

correctAnswer = 2;

break;

case 13:

pictureBox1.Image = Properties.Resources.Question\_13;

lblQuestion.Text = " Având graful ponderat din figura 4, să se determine un arbore de acoperire de cost minim. Vom considera nodul A ca nod de plecare. Care este ordinea de parcurgere a nodurilor?";

button1.Text = "A, B, D, E, C;";

button2.Text = "A, C, B, D, E;";

button3.Text = "A, B, D, C, E;";

button4.Text = "A, B, E, B, D;";

correctAnswer = 1;

break;

case 14:

pictureBox1.Image = Properties.Resources.Question\_14;

lblQuestion.Text = " Care din urmatoarele interpretări fac adevărată formula F, ilustrată în graficul din Figura 5?\r\n";

button1.Text = "x1 = true, x2 = true, x3 = true";

button2.Text = "x1 = false, x2 = true, x3 = false";

button3.Text = "x1 = true, x2 = false, x3 = true";

button4.Text = "niciuna din variante nu este corectă";

correctAnswer = 2;

break;

case 15:

pictureBox1.Image = Properties.Resources.Question\_15;

lblQuestion.Text = " Să se indice ordinul de creștere al funcției f(n) cu următoarea relație de recurență:";

button1.Text = "f(n) ∈ O(n!)";

button2.Text = "f(n) ∈ O(n^2)";

button3.Text = "f(n) ∈ O(n^2)";

button4.Text = "f(n) ∈ O(n \* n!)";

correctAnswer = 1;

break;

case 16:

pictureBox1.Image = Properties.Resources.Question\_16;

lblQuestion.Text = " Ce se va afișa pe ecran în urma execuției următorului program?";

button1.Text = "6";

button2.Text = "3";

button3.Text = "nimic (programul nu se compilează)";

button4.Text = "1";

correctAnswer = 4;

break;

case 17:

pictureBox1.Image = Properties.Resources.Question\_17;

lblQuestion.Text = " Considerăm următorul program:\r\n Dacă în urma execuției liniilor 4 și 5 se afisează 2 respectiv 0xF004, ce se va afișa în urma execuției liniei 6?";

button1.Text = "0xF010";

button2.Text = "0xF006";

button3.Text = "0xF00A";

button4.Text = "0xF003";

correctAnswer = 3;

break;

case 18:

pictureBox1.Image = Properties.Resources.Question\_18;

lblQuestion.Text = " Se dă următorul algoritm:\r\n Care vor fi valorile vectorului a dupa terminarea pasului i = 5, a = (8, 6, 4, 2, 3, 5, 7)?";

button1.Text = "(2, 3, 4, 5, 8, 7, 6)";

button2.Text = "(8, 7, 6, 5, 4, 2, 3)";

button3.Text = "(2, 3, 4, 5, 6, 8, 7)";

button4.Text = "(8, 3, 4, 5, 6, 7, 8)";

correctAnswer = 3;

break;

case 19:

pictureBox1.Image = Properties.Resources.Question\_19;

lblQuestion.Text = " Ce va afișa următorul program?";

button1.Text = "Franklin";

button2.Text = "FDR";

button3.Text = "Franklin D. Roosevelt";

button4.Text = "Roosevelt";

correctAnswer = 2;

break;

case 20:

pictureBox1.Image = Properties.Resources.Question\_20;

lblQuestion.Text = " Analizând algoritmul de mai jos, precizați ce timp de execuție are.";

button1.Text = "O(logN)";

button2.Text = "O(2^N)";

button3.Text = "O(NlogN)";

button4.Text = "O(N)";

correctAnswer = 4;

break;

}

}

private void button5\_Click(object sender, EventArgs e)

{

Form2 f2 = new Form2(); // buton care accesează jocul PAC-MAN

f2.Show();

}

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

{

}

private void Form1\_FormClosing(object sender, FormClosingEventArgs e)

{

Application.Exit(); // odată închisă această fereastră, se închide

// toată aplicația

}

}

}

Form2.cs – jocul PAC-MAN

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Runtime.CompilerServices;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace Profesorul\_Pac\_Man

{

public partial class Form2 : Form

{

bool goup, godown, goleft, goright, isGameOver;

int score, playerSpeed, redGhostSpeed, yellowGhostSpeed, pinkGhostX, pinkGhostY;

private void button1\_Click(object sender, EventArgs e)

{

}

// dupa finalizarea jocului, acesta va reîncepe mereu, dacă dorește

// utilizatorul

public Form2()

{

InitializeComponent();

resetGame();

}

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

{

}

// trimiterea programului informația că utilizatorul folosește cel puțin

// una din tastele sus-jos-stânga-dreapta

private void keyisdown(object sender, KeyEventArgs e)

{

if(e.KeyCode == Keys.Up)

{

goup = true;

}

if (e.KeyCode == Keys.Down)

{

godown = true;

}

if (e.KeyCode == Keys.Left)

{

goleft = true;

}

if (e.KeyCode == Keys.Right)

{

goright = true;

}

}

// trimiterea programului informația că utilizatorul nu mai folosește

// nicio tastă care să-l deplaseze pe PAC-MAN

private void keyisup(object sender, KeyEventArgs e)

{

if (e.KeyCode == Keys.Up)

{

goup = false;

}

if (e.KeyCode == Keys.Down)

{

godown = false;

}

if (e.KeyCode == Keys.Left)

{

goleft = false;

}

if (e.KeyCode == Keys.Right)

{

goright = false;

}

// când utilizatorul a încheiat jocul și apasă ENTER,

// jocul e resetat

if(e.KeyCode == Keys.Enter && isGameOver == true)

{

resetGame();

}

}

// ce se întâmplă cu fiecare obiect atunci când jocul e pornit

private void mainGameTimer(object sender, EventArgs e)

{

txtScore.Text = "Score: " + score;

// se stabilește ce se întâmplă cu PAC-MAN atunci când utilizatorul

// vrea să-l deplaseze folosind săgețile corespunzătoare

// la fiecare schimbare de sens, se va modifica și imaginea

// lui PAC-MAN, astfel încât să existe concordanță între

// sensul de mers și orientarea feței lui PAC-MAN

if(goleft == true)

{

pacman.Left -= playerSpeed;

pacman.Image = Properties.Resources.left;

}

if (goright == true)

{

pacman.Left += playerSpeed;

pacman.Image = Properties.Resources.right;

}

if (godown == true)

{

pacman.Top += playerSpeed;

pacman.Image = Properties.Resources.down;

}

if (goup == true)

{

pacman.Top -= playerSpeed;

pacman.Image = Properties.Resources.Up;

}

// când PAC-MAN depășește o margine, el va ajunge în partea opusă a // ecranului (dreapta-stânga și sus-jos)

if(pacman.Left < -10)

{

pacman.Left = 680;

}

if(pacman.Left > 680)

{

pacman.Left = -10;

}

if(pacman.Top < -10)

{

pacman.Top = 550;

}

if(pacman.Top > 550)

{

pacman.Top = 0;

}

// ce se întamplă cu fiecare obiect:

// ---- bănuțul colectat devine invizibil si se incrementează scorul

// utilizatorului

// ---- atingerea unui perete sau a unei fantome

// duce la pierderea jocului

foreach(Control x in this.Controls)

{

if(x is PictureBox)

{

if( (string)x.Tag == "coin" && x.Visible == true )

{

if(pacman.Bounds.IntersectsWith(x.Bounds))

{

score += 1;

x.Visible = false;

}

}

if( (string)x.Tag == "wall" )

{

if (pacman.Bounds.IntersectsWith(x.Bounds))

{

gameOver("Ai pierdut!");

}

// când fantoma roz atinge un zid, i se modifică

// sensul de mișcare pe axa Ox

if(pinkGhost.Bounds.IntersectsWith(x.Bounds))

{

pinkGhostX = -pinkGhostX;

}

}

if( (string)x.Tag == "ghost" )

{

if(pacman.Bounds.IntersectsWith(x.Bounds))

{

gameOver("Ai pierdut!");

}

}

}

}

// mișcarea fantomelor

// fantoma roșie și cea galbenă se deplasează de la stânga la // dreapta (pe axa Ox), între ziduri

redGhost.Left += redGhostSpeed;

if(redGhost.Bounds.IntersectsWith(pictureBox1.Bounds) || redGhost.Bounds.IntersectsWith(pictureBox2.Bounds))

{

redGhostSpeed = -redGhostSpeed; // atingerea unui zid = // schimbarea sensului de mișcare

// pe axa Ox

}

yellowGhost.Left -= yellowGhostSpeed;

if (yellowGhost.Bounds.IntersectsWith(pictureBox3.Bounds) || yellowGhost.Bounds.IntersectsWith(pictureBox4.Bounds))

{

yellowGhostSpeed = -yellowGhostSpeed;

}

pinkGhost.Left -= pinkGhostX;

pinkGhost.Top -= pinkGhostY;

// fantoma roz se deplasează în zig-zag (compunerea forțelor reprezentate pe Ox și Oy)

// daca atinge marginea ecranului, i se modifică sensul de mișcare

// pe axa Oy

if(pinkGhost.Top < 0 || pinkGhost.Top > 520)

{

pinkGhostY = -pinkGhostY;

}

if(pinkGhost.Left < 0 || pinkGhost.Left > 620)

{

pinkGhostX = -pinkGhostX;

}

if (score == 34)

{

gameOver("Ai Castigat!"); // încheierea jocului din ipostaza de

// câștigător

}

}

// la fiecare nou început de joc, se resetează scorul, respectiv vitezele // și coordonatele lui PAC-MAN și ale fantomelor

private void resetGame()

{

txtScore.Text = "Score: 0";

score = 0;

redGhostSpeed = 5;

yellowGhostSpeed = 5;

playerSpeed = 8;

pinkGhostX = 5;

pinkGhostY = 5;

isGameOver = false;

pacman.Left = 24;

pacman.Top = 41;

yellowGhost.Left = 476;

yellowGhost.Top = 449;

redGhost.Left = 191;

redGhost.Top = 55;

pinkGhost.Left = 566;

pinkGhost.Top = 238;

// în plus, fiecare bănuț redevine vizibil

foreach(Control x in this.Controls)

{

if(x is PictureBox)

{

x.Visible = true;

}

}

gameTimer.Start(); // jocul pornește

}

private void gameOver(string message) // mesajul afișat la sfârșitul // fiecărui joc

{

isGameOver = true;

gameTimer.Stop(); // jocul se oprește

txtScore.Text = "Score: " + score + Environment.NewLine + message;

}

}

}