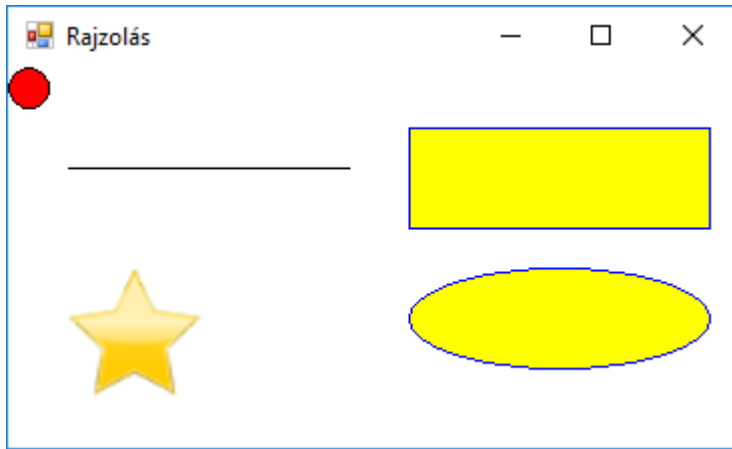


Programozás C# nyelven

9. előadás



<http://e-learning.ujs.sk/>



Paint esemény: akkor következik be, amikor az ablak kirajzolódik, pl. alkalmazás indításakor, ablak előtérbe kerülésekor, átméretezésekor, stb.

Az ablak „**Invalidate()**” metódusa segítségével bármikor kikényszeríthető az ablak újrarajzolása.

```
using System;
using System.Drawing;
using System.Windows.Forms;

namespace _034_Grafika
{
    public partial class Form1 : Form
    {
        // korvanal szinek
        private Pen kek = new Pen(Color.Blue);
        private Pen fekete = new Pen(Color.Black);
        // kitoltes szinek
        private SolidBrush sarga = new SolidBrush(Color.Yellow);
        private SolidBrush piros = new SolidBrush(Color.Red);
        // pontok
        private Point p1 = new Point(30, 50);
        private Point p2 = new Point(170, 50);
        private Point p3 = new Point(30, 100);
        // alakzatok
        private Rectangle teglalap = new Rectangle(200, 30, 150, 50);
        private Rectangle ellipszis = new Rectangle(200, 100, 150, 50);
        private Rectangle kor = new Rectangle(0, 0, 20, 20);
    }
}
```

```
public Form1()
{
    InitializeComponent();
}

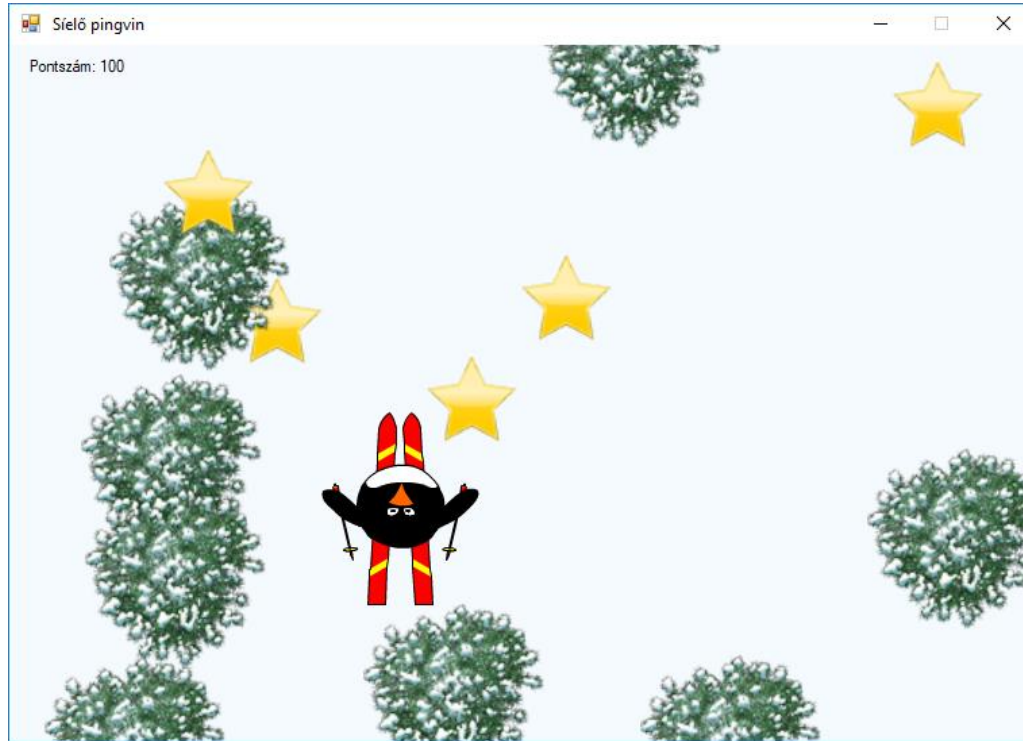
private void Form1_Load(object sender, EventArgs e)
{
    this.BackColor = Color.White;
    this.DoubleBuffered = true;
}

private void Form1_Paint(object sender, PaintEventArgs e)
{
    Graphics g = e.Graphics;
    // Rajzolas...
    g.DrawLine(fekete, p1, p2);
    g.DrawImage(Properties.Resources.csillag, p3);
    g.FillRectangle(sarga, teglalap);
    g.DrawRectangle(kek, teglalap);
    g.FillEllipse(sarga, ellipszis);
    g.DrawEllipse(kek, ellipszis);
    g.FillEllipse(piros, kor);
    g.DrawEllipse(fekete, kor);
}
```

```
private void timer1_Tick(object sender, EventArgs e)
{
    // kor objektom modositasa
    kor.X += 4;
    kor.Y += 2;
    kor.Width++;
    kor.Height++;
    // ujrarajzolas kikenyszeritese
    this.Invalidate();
}
```

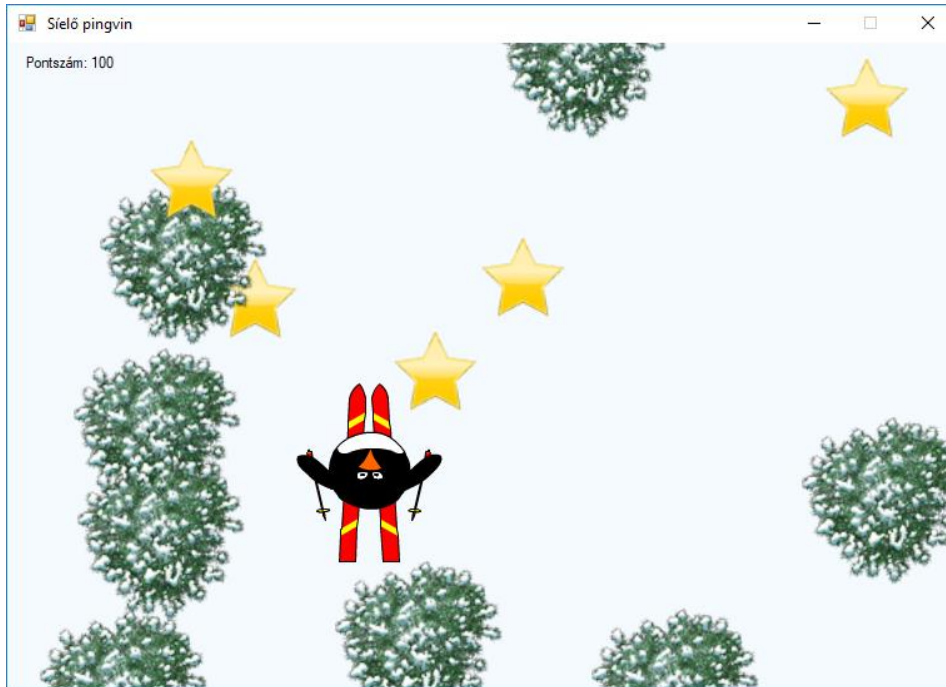
„Sielő pingvin” játék:

035 Sielo pingvin jatek



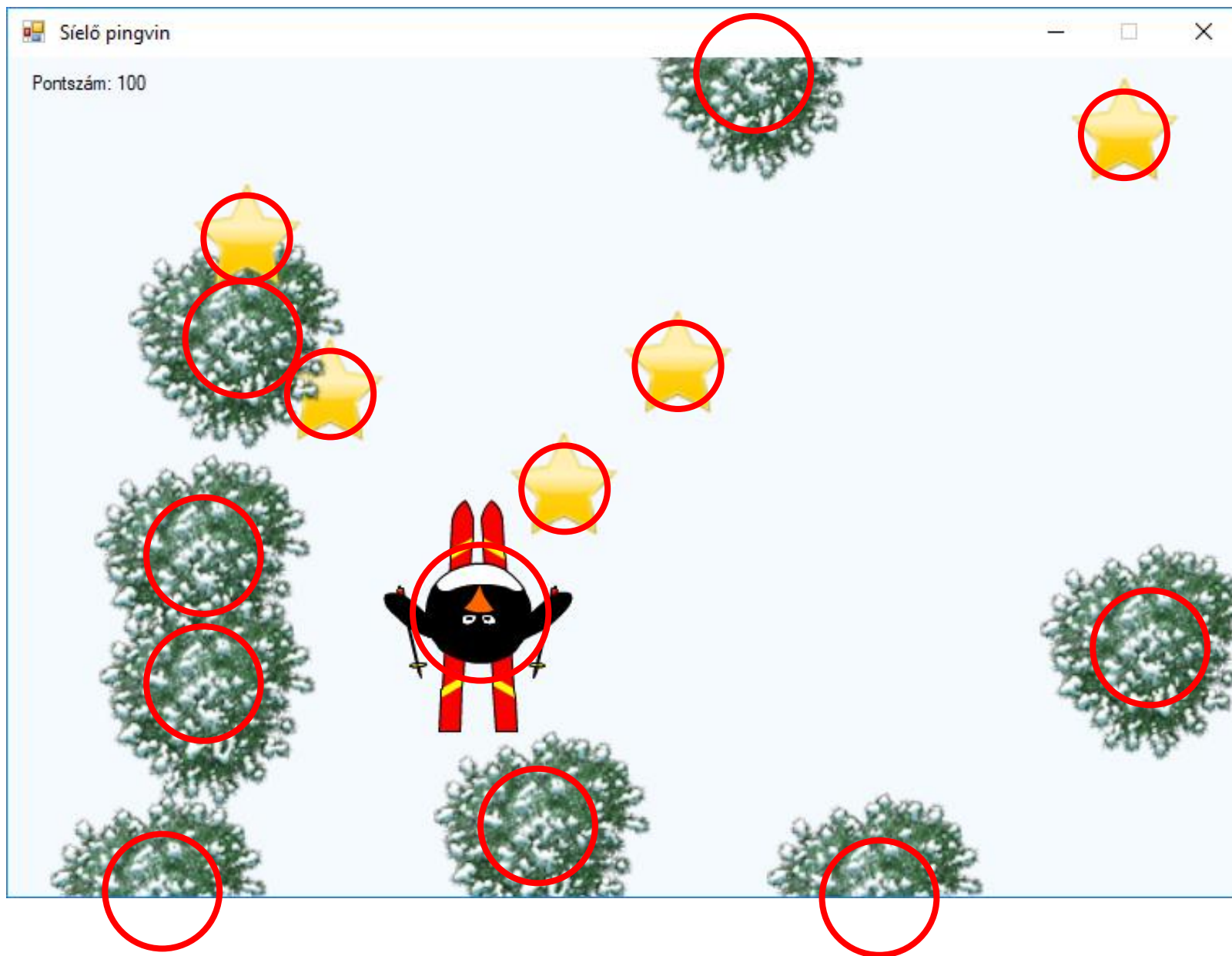
- Az objektumok (fenyőfák, csillagok) lefele „mozognak”.
- A pingvin az egér segítségével mozgatható vízszintesen jobbra-balra.
- Ha a pingvin nekimegy a fának: -100 pont.
- Ha a pingvin összeszed egy csillagot: +10 pont.

A játék megvalósítása:

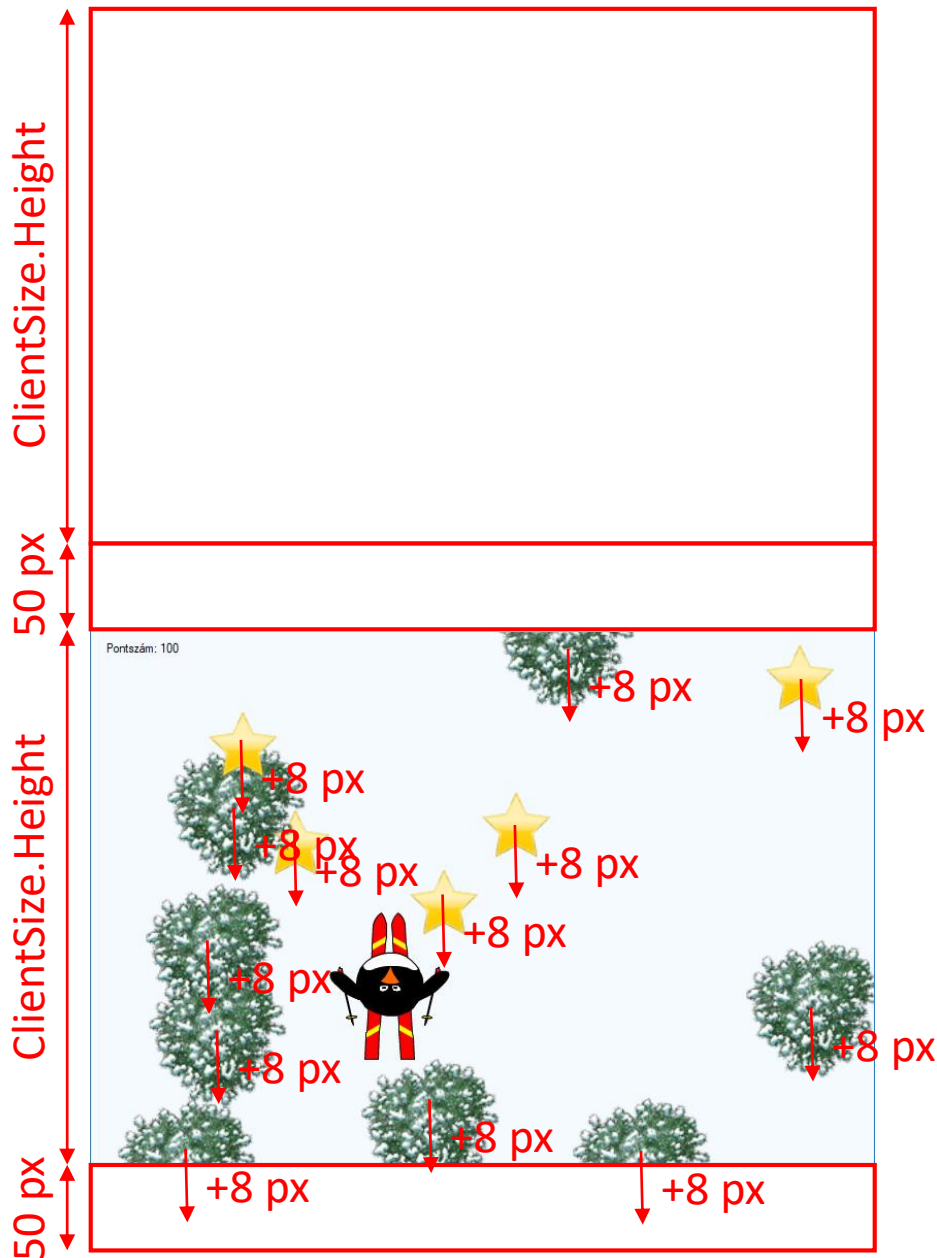


- Milyen objektumaink lesznek a játékban?
- Ezek közül melyek hasonlóak, melyek különbözőek egymástól?
- Mit és hogyan lehessen elvégezni az egyes objektumokkal?
- Ezek alapján az objektumok milyen adatmezőkkel és metódusokkal rendelkezzenek? Lehet öröklődést használni?
- Hogyan érzékeljük a programban az ütközést, amikor a pingvin összeszed egy csillagot vagy nekimegy egy fának?

Ütközés érzékelése:

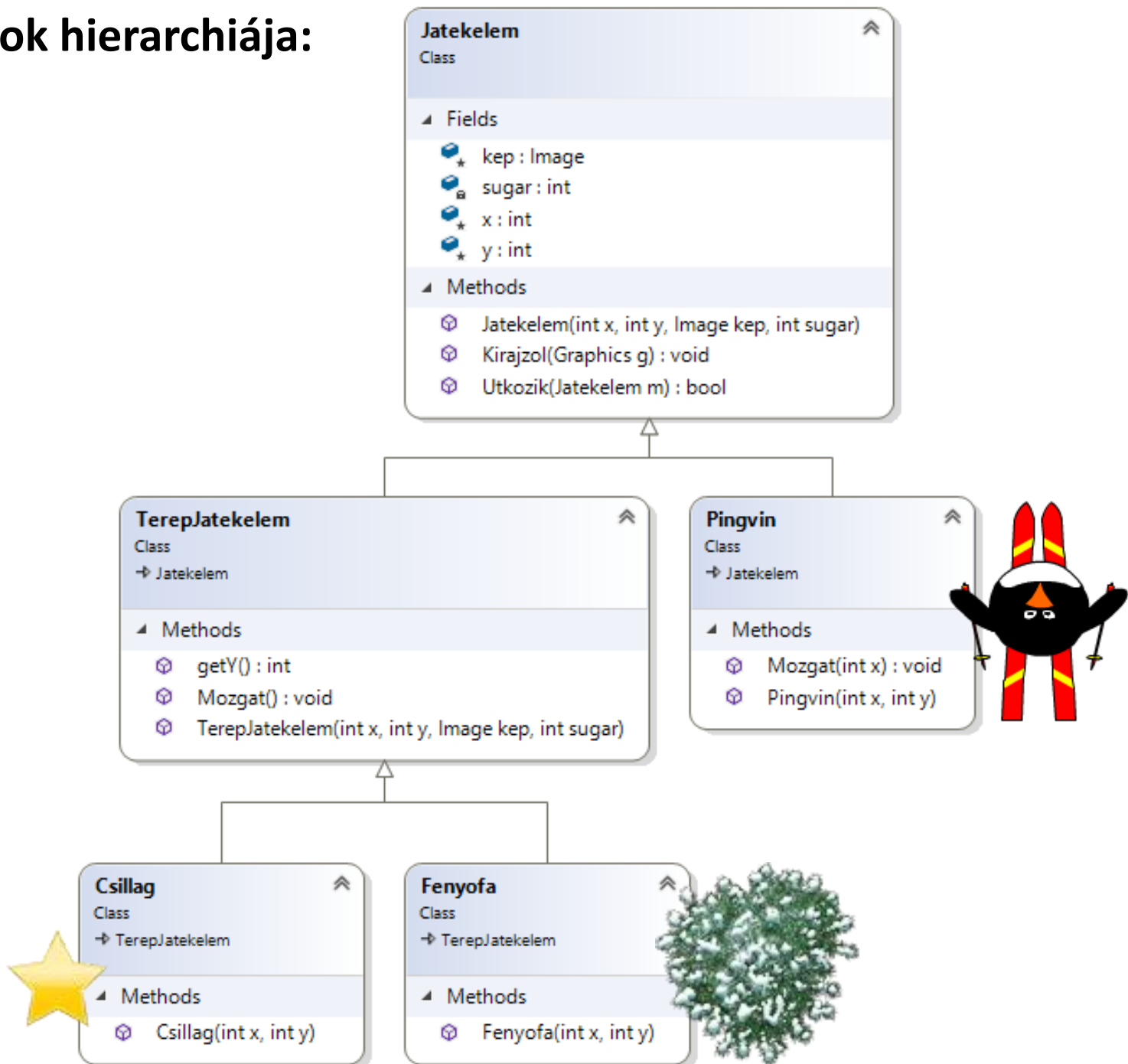


Objektumok mozgása:



- 15 darab objektum (fenyőfa vagy csillag)
- **Timer** (Interval: 40 ms)
 - **Tick** esemény: +8 px
- Játékos mozgása a **MouseMove** esemény segítségével (csak az **X** koordinátát használva)

Osztályok hierarchiája:



```
public class Jatekelem
{
    protected int x, y;    // koordinatak (kep kozeppontjahoz viszonyitva)
    protected Image kep;   // kirajzolando kep
    private int sugar;     // kozeppontol szamitott sugar (utkozes vizsgalatahoz)

    public Jatekelem(int x, int y, Image kep, int sugar)
    {
        this.x = x;
        this.y = y;
        this.kep = kep;
        this.sugar = sugar;
    }

    public void Kirajzol(Graphics g)
    {
        g.DrawImage(kep, x - kep.Width / 2, y - kep.Height / 2);
    }

    public bool Utkozik(Jatekelem m)
    {
        if (m==null)
        {
            return false;
        }
        double tav = Math.Sqrt(Math.Pow(this.x - m.x, 2) + Math.Pow(this.y - m.y, 2));
        return tav < this.sugar + m.sugar;
    }
}
```

```
public class TerepJatekelem : Jatekelem
{
    public TerepJatekelem(int x, int y, Image kep, int sugar) : base(x, y, kep, sugar)
    {
    }

    public void Mozgat()
    {
        y = y + 8;
    }

    public int getY()
    {
        return y;
    }
}
```

```
public class Csillag : TerepJatekelem
{
    public Csillag(int x, int y) : base(x, y, Properties.Resources.csillag, 20)
    {
    }
}
```

```
public class Fenyofa : TerepJatekelem
{
    public Fenyofa(int x, int y) : base(x, y, Properties.Resources.fenyofa, 30)
    {
    }
}
```

```
public class Pingvin : Jatekelem
{
    public Pingvin(int x, int y) : base(x, y, Properties.Resources.pingvin, 40)
    {
    }

    public void Mozgat(int x)
    {
        this.x = x;
    }
}
```

```
private Pingvin jatekos;  
private TerepJatekelem[] objektumok = new TerepJatekelem[15];  
private int pontszam = 0;  
private Random randgen = new Random();
```

```
private void Form1_Load(object sender, EventArgs e)
```

```
{
```

```
    // ablak beallitasa
```

```
    this.BackColor = Color.FromArgb(245, 250, 255);
```

```
    this.FormBorderStyle = FormBorderStyle.FixedSingle;
```

```
    this.MaximizeBox = false;
```

```
    this.DoubleBuffered = true;
```

```
    this.CenterToScreen();
```

```
    // egerkurzor elrejtese
```

```
    Cursor.Hide();
```

```
    // jatekos letrehozasa
```

```
    jatekos = new Pingvin(this.ClientSize.Width / 2,  
                          this.ClientSize.Height * 2/3 );
```

```
    // objektumok letrehozasa
```

```
    for (int i = 0; i < objektumok.Length; i++)
```

```
    {
```

```
        objektumok[i] = TerepJatekelemGeneralas(-this.ClientSize.Height - 50,  
                                                  this.ClientSize.Height + 50);
```

```
    }
```

```
}
```

```
private TerepJatekelem TerepJatekelemGeneralas(int yMin, int yMax)
{
    TerepJatekelem o;
    bool vanUtkozes;
    do
    {
        // létrehozunk egy objektumot
        if (randgen.Next(2) == 0)
        {
            // fa
            o = new Fenyofa(randgen.Next(this.ClientSize.Width),
                            randgen.Next(yMin, yMax));
        }
        else
        {
            // csillag
            o = new Csillag(randgen.Next(this.ClientSize.Width),
                            randgen.Next(yMin, yMax));
        }
        // megenzzuk utkozik-e a pingvinnel vagy valamelyik másik objektummal
        vanUtkozes = o.Utkozik(jatekos);
        for (int i = 0; i < objektumok.Length; i++)
        {
            if (o.Utkozik(objektumok[i]))
            {
                vanUtkozes = true;
            }
        }
    } while (vanUtkozes);
    return o;
}
```

```
private void Form1_Paint(object sender, PaintEventArgs e)
{
    Graphics g = e.Graphics;
    jatekos.Kirajzol(g);
    for (int i = 0; i < objektumok.Length; i++)
    {
        objektumok[i].Kirajzol(g);
    }
}
```

```
private void Form1_MouseMove(object sender, MouseEventArgs e)
{
    jatekos.Mozgat(e.X);
}
```

```
private void Form1_KeyDown(object sender, KeyEventArgs e)
{
    if (e.KeyCode == Keys.Escape)
    {
        this.Close();
    }
}
```



```
private void timer1_Tick(object sender, EventArgs e)
{
    this.BackColor = Color.FromArgb(245, 250, 255);
    for (int i=0; i<objektumok.Length; i++)
    {
        objektumok[i].Mozgat();
        if (objektumok[i].getY()>this.ClientSize.Height+50)
        {
            objektumok[i] = TerepJatekelemGeneralas(-this.ClientSize.Height - 50, -50);
        }
        if (jatekos.Utkozik(objektumok[i]))
        {
            if (objektumok[i] is Fenyofa)
            {
                pontszam -= 100;
                if (pontszam<0)
                {
                    pontszam = 0;
                }
                this.BackColor = Color.Red;
            } else
            {
                pontszam += 10;
                objektumok[i] = TerepJatekelemGeneralas(-this.ClientSize.Height - 50, -50);
            }
            label1.Text = "Pontszám: " + pontszam;
        }
    }
    this.Invalidate();
}
```

Összefoglalás:

- Grafika („Paint” esemény, „Graphics” osztály)
- „Síelő pingvin” játék megvalósítása

Ötletek a játék továbbfejlesztéséhez:

- Játék teljes képernyőn, több objektummal
- A pingvin lassabb jobbra-balra mozgatása
- A síelés gyorsítása/lassítása az egér fel-le mozgatásával
- Animált csillag, fa, pingvin megvalósítása Timer segítségével
- További játékelemek hozzáadása (pl. hóember)
- Életek hozzáadása a játékhoz
 - Ha fának ütközik a pingvin, egy élet levonása
 - Ha elfogynak az életek, vége a játéknak
- ...