

МІНІСТЕРСТВО ОСВІТИ ТА НАУКИ УКРАЇНИ  
ЛЬВІВСЬКИЙ  
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ імені ІВАНА ФРАНКА

**Лабораторна робота №4**  
з курсу «Паралельні та розподілені обчислення»

Виконав:  
Гуменюк Станіслав  
Група Пмі-33с

Оцінка \_\_\_\_  
Перевірив:  
Пасічник  
Т.В.

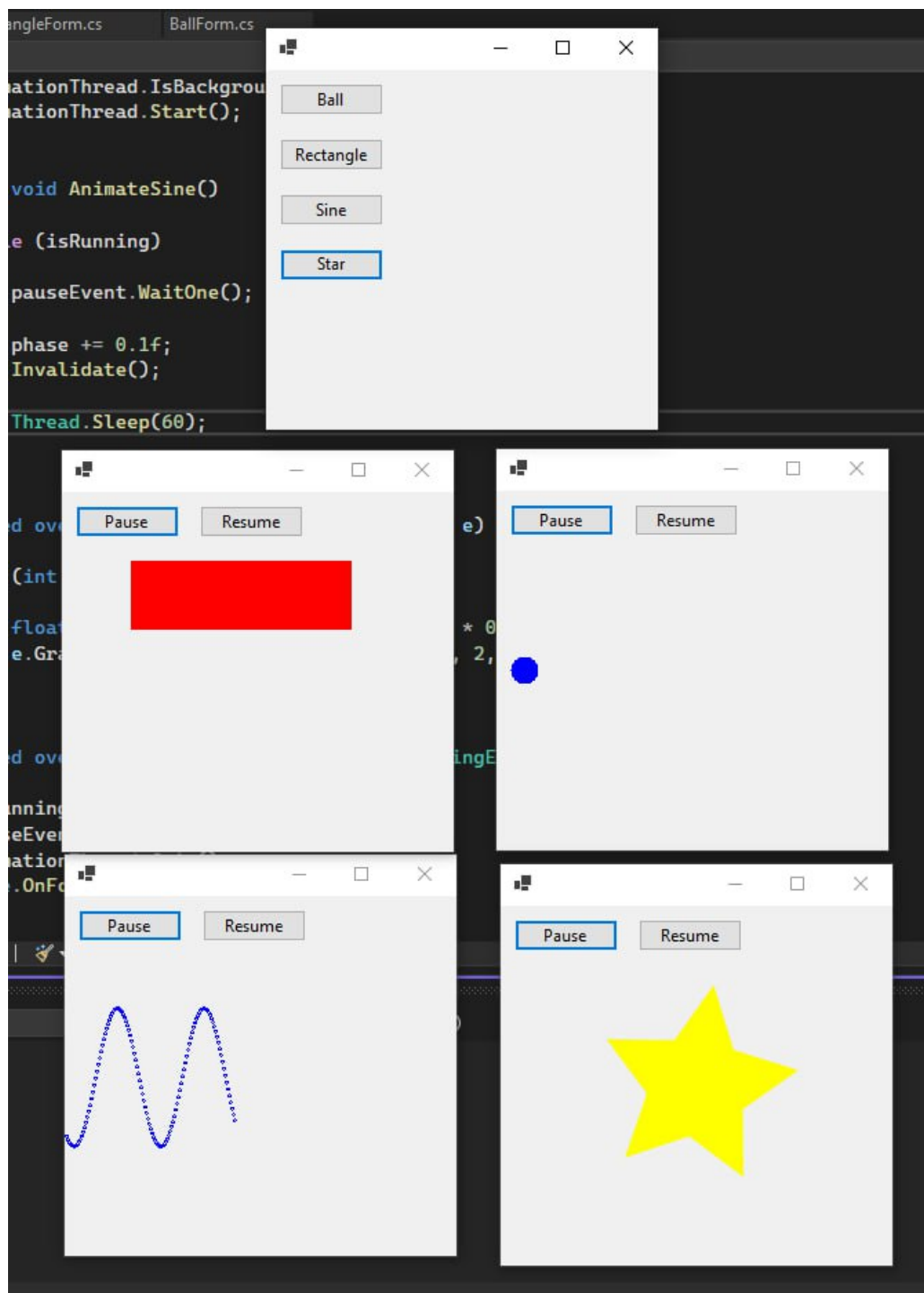
2024

## Завдання:

Створити 4 потоки, в кожному з яких зробити графічне представлення його роботи. Тобто в одному потоці літає кулька по заданій траєкторії, в іншому прямокутник змінює свої розміри, ще в іншому малюється синусоїда тощо. Ці графічні представлення повинні показувати різні (а не однотипні -  $\sin$ ,  $\cos$ ,  $\tan$ ) процеси та чітко відображати безперервну роботу потоку. В кожній з цих 4-х форм створити дві обов'язкові кнопки: одна призупиняє потік (не завершує!), інша – відновлює роботу потоку з точки зупинки. Ще можливі (за бажанням) кнопки завершення потоку та відкривання потоку (у вікні головного потоку).

## Програмна реалізація:

Програма написана на пакеті .NET з використанням мови C# та WinForms. В головній формі є чотири кнопки для запуску потоків з фігурами



На прикладі однієї з фігур покажу реалізацію потоку

Ініціалізація форми:

```
public BallForm()
{
    pauseEvent = new ManualResetEvent(true);
    isRunning = true;
    x = 50; y = 50; dx = 5; dy = 5;

    btnPause = new Button() { Text = "Pause", Location = new Point(10, 10) };
    btnResume = new Button() { Text = "Resume", Location = new Point(100, 10) };

    btnPause.Click += (sender, args) => pauseEvent.Reset();
    btnResume.Click += (sender, args) => pauseEvent.Set();

    Controls.Add(btnPause);
    Controls.Add(btnResume);

    animationThread = new Thread(AnimateBall);
    animationThread.IsBackground = true;
    animationThread.Start();
}
```

Функція для анімації:

```
private void AnimateBall()
{
    while (isRunning)
    {
        pauseEvent.WaitOne();

        x += dx; y += dy;
        if (x < 0 || x > ClientSize.Width - 20) dx = -dx;
        if (y < 0 || y > ClientSize.Height - 20) dy = -dy;

        Invalidate();

        Thread.Sleep(60);
    }
}
```

Функція для коректного завершення потоку при закритті форми:

```
protected override void OnFormClosing(FormClosingEventArgs e)
{
    isRunning = false;
    pauseEvent.Set();
    animationThread.Join();
    base.OnFormClosing(e);
}
```

Функція для відмалювання фігури

```
protected override void OnPaint(PaintEventArgs e)
{
    e.Graphics.FillEllipse(Brushes.Blue, x, y, 20, 20);
}
```