

## ЛАБОРАТОРНА РОБОТА 5

### Програмування засноване на тестуванні

**Мета роботи:** навчитися розробляти й реалізовувати модульні тести з використанням систем автоматизованого тестування.

**Завдання:**

1. Вивчити теоретичні відомості.
2. Ознайомитися із принципами роботи системи автоматизованого тестування.
3. Реалізувати модульні тести, що забезпечують перевірку основної функціональності кожного класу відповідно до розробленого проекту програмного продукту.
4. Домогтися успішного виконання всіх модульних тестів.
5. Зробити висновки з проекту програмного продукту

**Хід роботи:**

**1) Індивідуальне завдання:**

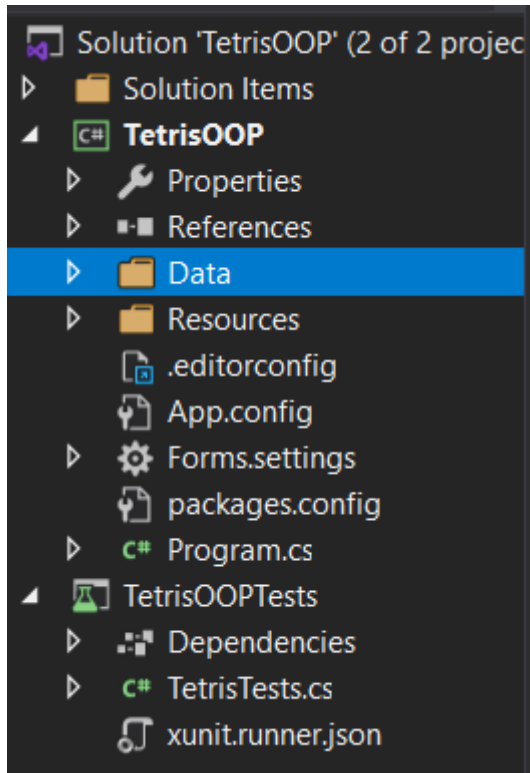
Тема: розробка однокористувацької двовимірної гри "Тетріс" з використанням принципів ООП.

"Тетріс" є головоломкою, побудованою на використанні геометричних фігур "тетраміно" - різновиду поліміно, що складаються з чотирьох квадратів.

Випадкові фігурки тетраміно падають зверху в прямокутний стакан шириною 10 і висотою 20 клітин. У польоті гравець може повертати фігурку та рухати її по горизонталі. Також можна «скидати» фігурку, тобто прискорювати її падіння, коли вже вирішено, куди фігурка повинна впасти. Фігурка летить, поки не наткнеться на іншу фігурку або на дно склянки. Якщо при цьому заповнився горизонтальний ряд з 10 кліток, він пропадає і все, що вище нього, опускається на одну клітку.

У спеціальному полі гравець бачить фігурку, яка буде слідувати після поточної — ця підказка дозволяє планувати свої дії. Темп гри поступово збільшується. Назва гри походить від кількості клітин, з яких складається кожна фігура. Гра закінчується, коли нова фігурка не може поміститися в стакан. Гравець отримує бали за кожен фігурку, тому його задача — заповнювати ряди, не заповнюючи саму склянку якомога довше, щоб таким чином отримати якомога більше балів.

## 2) Структура рішення за робочим та тестовим проектом:



## 3) Код юніт-тестів з використанням фреймворку xUnit:

```
using System;
using System.Collections.Generic;
using System.Text;
using TetrisOOP;
using Xunit;
using Engine;
using GameTetris;
using static Engine.EngineClass;
using TetrisOOP.Data.Modules.Users;
using System.Threading.Tasks;
using System.Net;

namespace TetrisOOP.Tests
{
    public class TetrisTests
    {
        [Fact]
        public void RandomFigure()
        {
            // Arrange
            RandomBag r = new RandomBag(1, 8);
            // Act
            int randomFigureId = r.Next();
            // Assert
            Assert.InRange(randomFigureId, 1, 8);
        }

        [Fact]
        public void CreateBoard()
        {
            // Arrange
            GameBoard gb = new GameBoard(5, 5);
            // Act
            CellType a = gb._cells[3, 3];
            // Assert
        }
    }
}
```

[illegible]

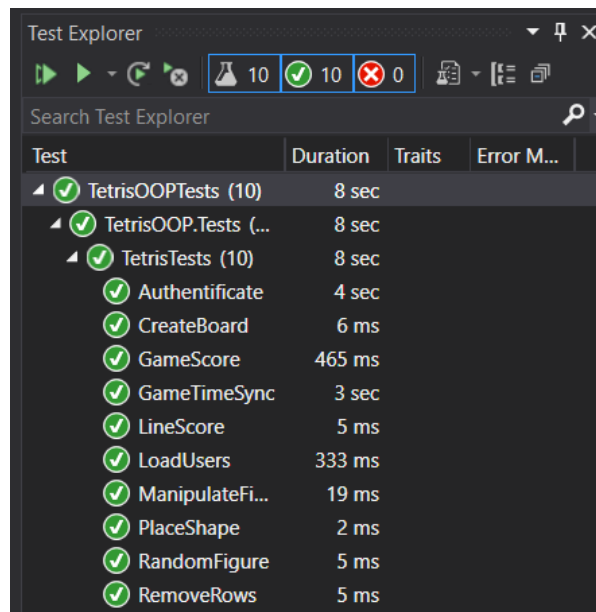
[illegible]

```

    // Arrange
    GameForm gf = new GameForm();
    // Act
    int gameScore = gf._game.Score;
    int defaultScore = Properties.Game.Default.CountScore;
    // Assert
    Assert.Equal(gameScore, defaultScore);
}
}
}

```

#### 4) Результати проведених тестів:



The screenshot shows the Test Explorer window with a summary bar at the top indicating 10 tests passed, 10 succeeded, and 0 failed. Below the summary, a table lists the test results for 'TetrisOOPTests (10)' and its sub-items.

Test	Duration	Traits	Error M...
✓ TetrisOOPTests (10)	8 sec		
✓ TetrisOOP.Tests (...)	8 sec		
✓ TetrisTests (10)	8 sec		
✓ Authenticate	4 sec		
✓ CreateBoard	6 ms		
✓ GameScore	465 ms		
✓ GameTimeSync	3 sec		
✓ LineScore	5 ms		
✓ LoadUsers	333 ms		
✓ ManipulateFi...	19 ms		
✓ PlaceShape	2 ms		
✓ RandomFigure	5 ms		
✓ RemoveRows	5 ms		

**Висновки:** я навчився розробляти й реалізовувати модульні тести з використанням системи автоматизованого тестування. Вивчив теоретичні відомості. Ознайомився із принципами роботи системи автоматизованого тестування xUnit для платформи .NET. Реалізував модульні тести, що забезпечують перевірку основної функціональності кожного класу, відповідно до розробленого проекту програмного продукту. Домігся успішного виконання всіх модульних тестів.