

Лабораторна робота №1 (максимально - 10 балів)

Тема: Повторення ООП. Гра “Хрестики-Нулики”

Мета роботи: згадати принципи ООП, написати цікаву консольну гру.

Завдання на лабораторну роботу

Завдання 0: Підготовка до виконання завдання

1. Встановити [DotNet CLI](#) 6 або 7 версії.
2. Створити окремий репозиторій на GitHub або GitLab. В цьому репозиторії будуть міститися всі виконані Вами завдання курсу “Конструювання програмного забезпечення”. Кожне завдання буде міститися в окремій директорії `lab-1`, `lab-2` ... `lab-n`
3. Надати доступ викладачу `vkich7`
4. Склонувати створений репозиторій.
5. Створити директорію `lab-1`
6. Зайти в створену директорію і запустити команду `dotnet new console`
7. Перейти до Завдання 1

Завдання 1: Створення гри “Хрестики-Нулики”

1. Ознайомитися зі специфікацією гри та додатковими фічами. Вам потрібно буде реалізувати як мінімум ОДНУ додаткову фічу на Ваш вибір.
2. Продумати структуру, можливі проблеми та труднощі, пов’язані з реалізацією та підтримкою коду гри.
3. Створити гру.
4. Зверніть увагу, що необов’язково реалізовувати УСІ додаткові фічі. Але це буде плюсом, якщо Ви обдумаєте, як би Ви їх реалізовували, і ще більшим плюсом, якщо реалізуєте більше однієї.
5. П’ять балів з можливих десяти Ви отримаєте за вчасну здачу. Ще 5 – за правильність реалізації.

Специфікація гри “Хрестики-Нулики”

1. Початковий ігровий екран має виглядати наступним чином:

```
PS E:\projects\tic-tac-toe> dotnet run
Let's play Tic Tac Toe!
Player 1: X
Player 2: O
```

Player 1's turn. Select from 1 to 9 from the game board.

```
 1 | 2 | 3
---+---+---
 4 | 5 | 6
---+---+---
 7 | 8 | 9
|
```

2. Кожна ігрова клітинка має свій номер. Якщо клітинка пуста, на екран виводиться її номер. Якщо клітинка заповнена, то на екран виводиться “X” або “O”:

```
PS E:\projects\tic-tac-toe> dotnet run
Let's play Tic Tac Toe!
Player 1: X
Player 2: O
```

Player 1's turn. Select from 1 to 9 from the game board.

```
 1 | 0 | X
---+---+---
 4 | X | 6
---+---+---
 7 | 0 | 9
```

3. Користувач обирає клітинку для заповнення, вводячи відповідні числа в консоль. Введені символи валідуються. Якщо введено число, якого немає в ігровому полі, виводиться повідомлення **There is no cell "{CHAR}" on the field.**

```
Let's play Tic Tac Toe!
Player 1: X
Player 2: O

Player 1's turn. Select from 1 to 9 from the game board.

 1 | 0 | X
---+---+---
 4 | X | 6
---+---+---
 7 | 0 | 9

There is no cell "10" on the field.
```

Якщо відповідна клітинка вже зайнята **Cell "{CHAR}" is already set.**

Якщо введено не число, то виводиться повідомлення **Please enter a valid number between 1 and 9.**

Повідомлення висвітлюється кожен раз поки користувач не введе прийнятний символ.

4. Після кожного ходу екран оновлюється так, що результати попередніх ходів зникають з екрану.

5. Гравці 1 і 2 ходять по чергову до тих пір поки один з них не виграє, або всі клітинки поля не будуть заповнені (нічийний результат).

6. В кінці гри потрібно показати результат:

- "Player 1 wins";
- "Player 2 wins";
- "It's a draw".

Після цього процес закінчується.

Додаткова фіча 1: Збереження та завантаження гри

Додайте можливість зберігати та завантажувати гру з файлу на комп'ютері.

1. Користувач в будь-який момент гри може натиснути кнопку "S". Після цього поточний стан гри має зберегтися в поточну директорію (`Directory.GetCurrentDirectory()`) в спеціальний файл з будь-яким розширенням на Ваш вибір ***saved-game.(xml, csv, txt)***.
2. Для того щоб завантажити збережену гру, потрібно запустити її з прапорцем `--load-saved`, напр., `dotnet run --load-saved`
3. Якщо користувач запускає команду для завантаження гри, а файлу, згаданого вище ще немає, то має з'явитися застереження **No saved game was found!** і гра розпочинається з початку.

```
PS E:\projects\tic-tac-toe> dotnet run --load-saved
No saved game was found!
Let's play Tic Tac Toe!
Player 1: X
Player 2: O

Player 1's turn. Select from 1 to 9 from the game board.

  1 | 2 | 3
---+---+---
  4 | 5 | 6
---+---+---
  7 | 8 | 9
```

Додаткова фіча 2: Одиночний режим гри

Додайте можливість одиночного режиму гри.

1. Комп'ютерний інтелект має якомога точніше слідувати алгоритму гри, зображеному в [статті](#).
2. Для запуску одиночного режиму потрібно передати прапорець `--single-player`, напр., `dotnet run --single-player`

Додаткова фіча 3: Повторення гри

Додайте можливість повторити гру та ведення загального рахунку.

1. По завершенню гри, не виходьте з процесу, а запитайте користувача **Do you want to play again? (y/n)**
2. Якщо користувач обирає **"n"**, то гра завершується як в попередній версії.
3. Якщо користувач обирає **"y"**, то гра повторюється, але у користувачів змінюються їх значки (X => O), і виводиться інформація про загальний рахунок.

```
PS E:\projects\tic-tac-toe> dotnet run
Let's play Tic Tac Toe!

Player 1: O [2]
Player 2: X [3]

Player 1's turn. Select from 1 to 9 from the game board.

  1 | 2 | 3
  ---+---+---
  4 | 5 | 6
  ---+---+---
  7 | 8 | 9
```

Додаткова фіча 4: Скасування дії

Додайте можливість скасовувати попередню дію.

1. Користувач може натиснути кнопку **"U"** для скасування попередньої дії (X чи O зникне і відповідна клітинка очиститься).
2. Якщо гра запущена в одиночному режимі, то одним натисканням на **"U"** має скасуватися і дія комп'ютерного інтелекту.