

## Лабораторна робота №5

### Тема: Гра в хрестики-нолики. ООП

**Мета роботи:** Навчитися створювати класи в Python, що відповідають за різні аспекти гри "Хрестики-нулики". Поглибити розуміння ООП через практичне застосування.

### Завдання:

1. Створення класу для відображення ігрового поля.
2. Реалізація класу для управління ходами комп'ютера.
3. Розробка класу для обробки ходів користувача.

### ТЕОРЕТИЧНА ЧАСТИНА

Ваші ходи та ходи комп'ютера зберігайте в двовірному списку. Допрцуйте програму таким чином, щоб класи які відповідають за виконання дій вносили зміни в список, а клас який виводить дошку - читав значення зі списку.

Створіть окремий клас, який перевірятиме умову завершення гри.

### ПОРЯДОК ВИКОНАННЯ

1. Клас ігрового поля `Board` відповідає за відображення поля на екрані. Клас має зберігати стан ігрового поля і надавати метод `display`, який виводить поточний стан поля у консоль.

```
class Board:
    def __init__(self):
        # Ваш код тут: ініціалізація поля гри
        pass

    def display(self):
        # Ваш код тут: виведення поля гри на екран
        pass
```

2. Клас комп'ютерного гравця `ComputerPlayer` буде визначати логіку ходу комп'ютера. У цьому класі реалуйте найпростішу логіку для випадкового ходу в пусту клітинку.

```
class ComputerPlayer:
    def __init__(self, symbol):
        # Ваш код тут: ініціалізація комп'ютерного гравця
        pass

    def choose_move(self, board):
        # Ваш код тут: логіка вибору ходу
        # Підказка: використайте випадковість для вибору ходу на початкових етапах
        pass
```

3. Клас гравця `HumanPlayer` дозволяє користувачу вводити свої ходи з клавіатури. Методи класу повинні включати перевірку введених даних на коректність.

```
class HumanPlayer:
    def __init__(self, symbol):
        # Ваш код тут: ініціалізація гравця
        pass

    def make_move(self):
        # Ваш код тут: отримання ходу від користувача
        # Підказка: використайте `input()` для отримання даних від користувача
        pass
```

4. Клас `EndGame` на власний розсуд (я вірю в вас)

Кожен клас має містити коментарі, що описують логіку роботи методів та можливі місця для оптимізації коду або алгоритмів.

### ЗАВДАННЯ НА 90+

Стилізуйте виведення в консоль, додавши очищення та курування "Зіграти ще раз / вийти". Додайте кількість виграних чи програних ігор в даному сеансі.

### ЗАВДАННЯ ДЛЯ PYTHONMONSTERS

<https://drive.google.com/file/d/1FYBS3T06sob-Ey4PK5YMuglFfli6mq0D/view?usp=sharing>

PS

```
def print_board(board):
    """
    Функція для виведення ігрового поля.
    Матриця 3x3, де кожен елемент - це строка, що представляє клітинку поля:
        " " - пуста клітинка, "X" - хрестик, "O" - нулик.
    """
    print("-----")
    for row in board:
        print("|" + "|".join(row) + "|")
        print("-----")

# Створення пустого поля
board = [[" " for _ in range(3)] for _ in range(3)]

# Виведення пустого поля
print_board(board)
```