***Практична робота №6. Проектування і реалізація класу для роботи з динамічними матрицями***

***Мета:*** *закріплення практичних навичок перевантаження операторів*

***Актуалізація опорних знань***

1. Чим відрізняються динамічні масиви від статичних?
2. Які операції використовують для динамічного виділення пам’яті?
3. Що означає копіювання об’єкта?

***Завдання до практичної роботи***

1. Створити клас для роботи з цілочисельними **динамічними** матрицями *Matrix*. Закритими членами класу є *n*, *m*, *\*\*arr*.
2. У класі реалізувати наступні методи:

* конструктор з параметрами, у якому виділяти динамічну пам’ять;
* конструктор копіювання для копіювання полів одного об’єкта до іншого;
* деструктор, у якому вивільняти динамічну пам’ять;
* введення матриці з клавіатури (або рандомне заповнення масиву) та вивід матриці на екран.

1. У цьому ж класі, перевантажити оператори:

* + (додавання матриць);
* – (віднімання матриць);
* \* (множення матриць);
* \* (множення матриці на число).

1. Додатково реалізувати (на оцінку «відмінно»):

* пошук транспонованої матриці.

1. Програма має містити меню, яке дозволить перевірити основні функції, створені у програмі.
2. Також у програмі потрібно надати користувачу можливість введення розмірності матриці *n* та *m*. Забезпечити перевірку розмірності, залежно від обраної користувачем операції.
3. Продемонструвати роботу створеної програми.

***Методичні рекомендації***

**Правило трьох** («Закон великої трійки», «Велика трійка») – практичне правило в C++, яке каже, що якщо в класі визначений один з таких методів, то, найпевніше, в ньому мають бути визначені всі три:

1. Деструктор (destructor).
2. Конструктор копіювання (copy constructor).
3. Оператор присвоєння копіюванням (copy assignment operator).

***Контрольні запитання:***

1. Що таке конструктор копіювання?
2. Як перевантажити оператор присвоєння?
3. Чому іноді варто явно описати конструктор копіювання, деструктор та перевантажити оператор присвоєння?