

МОДУЛЬ 3*Лабораторна робота 3.2***ДОСЛІДЖЕННЯ КОЛЕКЦІЙ ТА УЗАГАЛЬНЕНИХ КОЛЕКЦІЙ C#**

Мета роботи – дослідити класи-колекції та узагальнені колекції у мові програмування C#.

Завдання

1. Описати клас, заданий варіантом у мові програмування C# (табл. 1).
2. Створити список об'єктів класу, вказаного в п.1, використовуючи будь-яку бібліотечну узагальнену (Generic), звичайну (non-generic) колекцію C#, а також масив. Продемонструвати основні операції з колекціями та масивами: додавання, видалення, оновлення, пошук елемента та прохід по набору даних. Порівняти поведінку та пояснити відмінності. Максимальна оцінка за виконання завдання 1-2 - «задовільно».
3. Створити бінарне дерево об'єктів класу, вказаного в п.1. При цьому для бінарного дерева передбачити можливість зберігати дані будь-якого типу (створити узагальнений тип). Елементами дерева не можуть бути структури. Максимальна оцінка за виконання завдання 1-3 - «добре».
4. Доповнити бінарне дерево з п.3 операцію порівняння (використати узагальнені типи та інтерфейси `IComparer<T>` і/чи `IComparable<T>`). Продемонструвати функціонал, реалізований з інтерфейсів порівняння через виклик методів сортування. А також реалізувати власний ітератор, використовуючи інтерфейси `IEnumerable` та `IEnumerator`. Обхід дерева реалізувати відповідно до віріанту з табл 1 (колонка «Порядок обходу дерева*»). Максимальна оцінки за виконання завдань 1-4 – «відмінно».

Варіанти

Табл. 1

Ном ер варі анта	Устрій класу			Порядок обходу дерева*
	дані	методи та властивості	кількість об'єктів	
1	Студент: Номер квитка, ПІБ, рік народження, Номер групи	Переведення на наступний курс; обчислення віку; виведення інформації про студента	3	Прямий порядок (preorder)
2	Книга: Серійний номер, назва, рік видання, вартість примірника, кількість примірників	Збільшення вартості на певний процент; загальна вартість тиражу; виведення інформації про книгу	4	Зворотній порядок (postorder)
3	Товар: Код, назва, виробник, вартість, кількість у партії	Загальна вартість партії; збільшення вартості на певний процент; виведення інформації про товар	5	Центрований (центральный) порядок (inorder)
4	Коло: Колір заповнення, колір контуру, радіус	Обчислення площі; обчислення окружності; виведення інформації про фігуру	6	Прямий порядок (preorder)
5	Прямокутник: Колір заповнення, колір контуру, довжини сторін	Обчислення площі; обчислення периметра; виведення інформації про фігуру	5	Зворотній порядок (postorder)
6	Трапеція: Колір заповнення, колір контуру, довжини основ, висота	Обчислення площі; обчислення периметра; виведення інформації про фігуру	4	Центрований (центральный) порядок (inorder)
7	Вектор:	Обчислення довжини вектора;	3	Прямий порядок (preorder)

Ном ер варі анта	Устрій класу			Порядок обходу дерева*
	дані	методи та властивості	кількість об'єктів	
	Колір лінії, координати кінця вектора	збільшення вектора на певний відрізок; виведення інформації про вектор		
8	Рядок: Значення, довжина	Пошук заданого символу; зміна порядку наступності на протилежний; додавання нового рядка до існуючого; виведення рядка	4	Зворотній порядок (postorder)
9	Вектор: Колір лінії, полярні координати (довжина і кут)	Зменшення вектора в певну кількість разів; збільшення вектора в певну кількість разів; виведення інформації про вектор	5	Центрований (центральний) порядок (inorder)
10	Рядок: Значення, довжина, закритий ключ шифрування (різниця кодів та напрямки)	Шифрування рядка(заміна значущих символів на символи з кодом, більшим або меншим на певне значення); дешифрування рядка; виведення рядка	6	Прямий порядок (preorder)
11	Рядок: Значення, довжина	Пошук підрядка; вставка підрядка; заміна підрядка; виведення рядка	5	Зворотній порядок (postorder)
12	Рахунок: Код власника, сума	Поповнення рахунку; зняття готівки; переведення на іншого власника; перерахування за певним курсом; виведення інформації про рахунок	4	Центрований (центральний) порядок (inorder)
13	Товар: Назва, код, дата виготовлення, термін придатності	Визначення придатності до споживання; визначення кінцевого строку реалізації; виведення інформації про товар	3	Прямий порядок (preorder)
14	Рядок: Значення, довжина	Отримання підрядка з цифрами; отримання підрядка з буквами; перегрупування символів: спочатку букви, потім цифри; виведення рядка	4	Зворотній порядок (postorder)
15	Рейс: №, дата, пункт вибуття, пункт прибуття, час виїзду, час прибуття	Тривалість рейсу; час, що минув після закінчення рейсу; виведення інформації по рейс	5	Центрований (центральний) порядок (inorder)

Контрольні запитання та завдання

1. Дайте визначення колекції. Наведіть типи колекцій FCL.
2. Що таке операції «упаковка» і «розпаковка»? Наведіть приклади.
3. Наведіть основні інтерфейси, які успадковуються колекціями, та їх призначення.
4. Наведіть класи колекцій, які реалізують типову поведінку основних структур даних.
5. Поясніть призначення нумератору (ітератору) колекцій.
6. Як реалізувати власний нумератор (ітератор)? В яких випадках він потрібний?
7. Дані якого формату зберігаються у хеш-таблиці?
8. Поясніть особливості узагальнених типів .Net.
9. Поясніть сутність обмежень узагальнених типів.
10. Що може бути реалізовано, як узагальнення в C#?
11. Порівняйте універсальні, узагальнені колекції та масиви.

12. Коли варто створювати власні класи колекцій?
13. В чому різниця між інтерфейсами `ICollection` та `ICollectionable`? Для чого вони призначені? В яких випадках варто використовувати кожний із них?
14. В чому різниця між універсальними інтерфейсами `ICollection`, `ICollectionable` та узагальненими `ICollection<T>`, `ICollectionable<T>`? Для чого призначені ці узагальнені інтерфейси? В яких випадках варто використовувати кожний із них?
15. Для чого призначені інтерфейси `IEnumerable`, `IEnumerator` та `IEnumerable<T>`, `IEnumerator<T>`?