

**МОДУЛЬ 3***Лабораторна робота 3.4***СТВОРЕННЯ ТА ВИКОРИСТАННЯ ДЕЛЕГАТИВ, ПОДІЙ ТА ЛЯМБДА-ВИРАЗІВ**

**Мета роботи** – дослідити механізм обробки подій у мові C#, навчитися створювати та використовувати делегати.

**Завдання**

1. Описати на мові C# лямбда-вираз чи анонімний метод згідно з варіантом та викликати його через делегата з відповідним підписом.\*

2. Створити компонент багаторазового використання (клас), що містить член-подію. Події-об'єкти відповідають деяким подіям реального об'єкта, що моделюється даним компонентом (наприклад, якщо компонент є моделлю об'єкта „автомобіль”, подіями можуть бути „початок руху”, „зупинка”, „закінчення пального” і т. ін.). Для події створити клас-аргумент події. Для опису події створити новий делегат або використати наявний делегат бібліотеки FCL (наприклад, EventHandler).\*\*

3. Використовуючи створений у п.3 компонент, створити додаток, у якому визначити метод-обробник події для цього компонента, що реалізує реакцію додатка на подію (наприклад, повідомлення користувачеві про виникнення події). Метод-обробник події повинний отримувати інформацію про об'єкт-ініціатор події та аргумент події.\*\*

**Варіанти**

Номер варіанта	Метод для виклику через делегата	Клас (функціональність)	Подія
1	Сортування одновимірного масиву цілих чисел за зростанням	Машина: Початок руху, рух, кінець руху, заправлення паливом, витрата палива тощо	Закінчення бензину при переміщенні
2	Пошук у рядку кількості використань заданого символу	Математичні операції: додавання, віднімання, множення чисел та ін. операції над значеннями	Цілочисельне ділення
3	Визначення номера першого заданого символу в рядку	Стек (обмежена ємність): ініціалізація стеку, розміщення та видалення даних зі стеку	Переповнення стеку
4	Сортування одновимірного масиву символів за спаданням	Черга (обмежена ємність): ініціалізація черги, розміщення та видалення даних з черги	Переповнення черги
5	Визначення кількості цифр у рядку	Стек: ініціалізація стеку, розміщення та видалення даних зі стеку	Очищення стеку
6	Визначення наявності літер у рядку	Черга: ініціалізація черги, розміщення та видалення даних з черги	Очищення черги
7	Визначення суми елементів двовимірного цілого масиву	Арифметика з цілими числами: додавання, віднімання, множення чисел та ін. операції над значеннями	Переповнення розрядної сітки
8	Отримання сум елементів рядків двовимірного масиву в одновимірний масив	Банківський рахунок: операції з банківськими рахунками (відкриття, поповнення, зменшення, закриття тощо)	«overdraft» чи перевищення кредиту
9	Порівняння відповідних значень двох масивів	Internet-послуга: вибір тарифної моделі, надання доступу користувачеві, поповнення рахунку, підсумки трафіку, користування Internet та ін.	Перевищення трафіку

Номер варіанта	Метод для виклику через делегат	Клас (функціональність)	Подія
10	Отримання зі значень головної діагоналі двовимірного рядка одновимірного рядка	Мобільний зв'язок: активізація пакету, поповнення рахунку, стан рахунку, облік часу розмов та ін.	Зменшення рахунку до мінімуму
11	Визначення кількості малих символів у рядку	Машина: початок руху, рух, кінець руху, заправлення паливом, витрата палива тощо	Перевищення максимально допустимої швидкості
12	Визначення кількості цифр, літер та інших символів у рядку	Басейн: завдання розмірів, обчислення об'єму води, доливання та злив води	Переповнення басейну
13	Визначення, яких символів (літер чи цифр), у рядку більше	Список: ініціалізація списку, розміщення та видалення даних з списку	Очищення списку
14	Визначення кількості заголовних символів у рядку	Математичні операції: додавання, віднімання, множення чисел та ін. операції над значеннями	Результат операції дорівнює нулю
15	Визначення номера останнього заданого символу в рядку	Машина: початок руху, рух, кінець руху, обчислення пробігу, витрата палива тощо	Перевищення максимальної довжини пробігу

\* Для оцінки «добре» реалізувати і анонімний метод, і лямбда-вираз

\*\* Для оцінки «відмінно» обов'язково повинні бути: 1) клас-аргумент подій, 2) клас, в якому генерується подія (знаходяться сама подія та метод, що її викликає), 3) клас (-и), в якому є метод (-и) обробники події, 4) метод-обробник події обов'язково повинні мати сигнатуру, як у стандартних подіях .NET Framework. Допускається використовувати generic EventHandler.

### Контрольні запитання та завдання

1. Роз'ясніть сутність поняття делегата.
2. Наведіть приклад опису делегата та виклику методу, використовуючи делегат.
3. Наведіть склад класу делегату.
4. Поясніть, чим забезпечується контроль типів в делегатах.
5. У що розгортається делегат? (чим насправді є делегат, із яких компонентів він складається)\*\*\*
6. Поясніть сутність поняття анонімного методу.
7. Поясніть сутність лямбда-виразу, наведіть приклади лямбда-виразів.
8. Поясніть побічні ефекти, що виникають при використанні анонімних методів та лямбда-виразів.\*\*\*
9. Наведіть приклад опису події.
10. Поясніть, яким чином генерується подія.
11. Поясніть, яким чином виконується підписання на події та скасування підписки.
12. Поясніть особливості мультиадресних делегатів.
13. Чим відрізняється делегат та подія?

\*\*\* Питання для оцінки «відмінно».