

# ООП на базі С# .NET

## I. Знайомство з платформою .NET та мовою С#

### Інформація:

1. Історія та мета створення платформи .NET та мови С#.
2. Поняття CLR, CIL, CTS, Assembly
3. Потенційні переваги та недоліки стеку технологій.
4. Найпростіша програма на С#.NET
5. Робота з консоллю (Console)

### Джерела:

1. Троелсен-6, 44-73, 78-86, 91-96, 100-102, 104-115
2. [https://mva.microsoft.com/ru/training-courses/-c-1-8669?l=MAuqZiG1\\_4404984382](https://mva.microsoft.com/ru/training-courses/-c-1-8669?l=MAuqZiG1_4404984382)
3. <https://docs.microsoft.com/en-us/dotnet/csharp/index>
4. Для базового рівня  
[https://mva.microsoft.com/ru/training-courses/-c-8622?l=DQMesDF1\\_7804984382](https://mva.microsoft.com/ru/training-courses/-c-8622?l=DQMesDF1_7804984382)

### Завдання:

1. Створити консольне застосування на С#, яке вводить ціле число та виводить його факторіал.

## II. Базовий синтаксис мови С# - частина 1

### Інформація:

1. Вбудовані типи.
2. Змінні
3. Керування потоком виконання (керуючі конструкції – if, switch, for, while, foreach)

### Джерела:

1. Троелсен-6, 116-145
2. [https://mva.microsoft.com/ru/training-courses/-c-1-8669?l=3QCRmiG1\\_6204984382](https://mva.microsoft.com/ru/training-courses/-c-1-8669?l=3QCRmiG1_6204984382)
3. [https://mva.microsoft.com/ru/training-courses/-c-1-8669?l=FHwAviG1\\_2804984382](https://mva.microsoft.com/ru/training-courses/-c-1-8669?l=FHwAviG1_2804984382)
4. Для базового рівня  
[https://mva.microsoft.com/ru/training-courses/-c-8622?l=qt9UFEF1\\_8004984382](https://mva.microsoft.com/ru/training-courses/-c-8622?l=qt9UFEF1_8004984382)

### Завдання:

1. Створити консольне застосування на С#, яке виводить на екран «м'ячик в кімнаті» - зірочка літає по вікну-консолі та відбивається від його стінок. Виглядати може приблизно так:  
<https://monosnap.com/file/NVUWsaxwOYaN199j9FY8ZYIVmBfQgF>
2. Додати другу «зірочку» - так, щоб обидві починали рух з випадкових позицій, і відбивались як від меж вікна, так і одна від одної.

### III. Базовий синтаксис мови С# - частина 2

#### Інформація:

1. Масиви
2. Методи (функції)
3. Структури
4. Варіанти передачі та повернення параметрів
5. Типи-посилання та типи-значення

#### Джерела:

1. Троелсен-6, 146-180
2. [https://mva.microsoft.com/ru/training-courses/-c-1-8669?l=vBmhoiG1\\_2004984382](https://mva.microsoft.com/ru/training-courses/-c-1-8669?l=vBmhoiG1_2004984382)
3. [https://mva.microsoft.com/ru/training-courses/-c-1-8669?l=1zIKGjG1\\_7304984382](https://mva.microsoft.com/ru/training-courses/-c-1-8669?l=1zIKGjG1_7304984382)
4. [https://mva.microsoft.com/ru/training-courses/-c-1-8669?l=RlrpljG1\\_8404984382](https://mva.microsoft.com/ru/training-courses/-c-1-8669?l=RlrpljG1_8404984382)

#### Завдання:

1. Замість двох «зірочок», програма при старті повинна запитати в користувача їх кількість, а далі створити масив структур, кожна з яких містить параметри однієї «зірочки», і реалізувати поведінку за тими самим правилами, що і в попередньому завданні.
2. Максимально розбити код на методи (поки що можна розміщувати всі методи в класі Program і робити статичними).

### IV. Основа ООП в С#

#### Інформація:

1. Класи – базовий синтаксис
2. Простори імен
3. Підключення залежностей
4. Розбиття проекту на файли
5. Конструктори

#### Джерела:

1. Троелсен-6, 182-194
2. [https://mva.microsoft.com/ru/training-courses/-c-1-8669?l=4SGxGjG1\\_6304984382](https://mva.microsoft.com/ru/training-courses/-c-1-8669?l=4SGxGjG1_6304984382)
3. [https://mva.microsoft.com/ru/training-courses/-c-1-8669?l=mGvitjG1\\_7704984382](https://mva.microsoft.com/ru/training-courses/-c-1-8669?l=mGvitjG1_7704984382)
4. Для базового рівня  
[https://mva.microsoft.com/ru/training-courses/-c-8622?l=5ZNraHF1\\_2204984382](https://mva.microsoft.com/ru/training-courses/-c-8622?l=5ZNraHF1_2204984382)

#### Завдання:

1. Переробити застосування з попереднього завдання з використанням класів:
  - а. Створити клас «м'яч», який, поки що, буде відображатися як та сама зірочка, і інкапсулювати в ньому функціонал відбивання та переміщення. Замість масиву структур використати масив класів, і замість методів класу Program, які приймали

змінну типу структури, реалізувати методи з аналогічною функціональністю всередині класу «м'яч».

- b. Передбачити, що «м'яч» може мати довільні розміри (як варіант, для реалізації поведінки можна задавати розміри параметром «радіус», а положення - як координати центру). Переробити функції відбивання так, щоб вони спиралися лише на нові параметри.

## V. ООП в С# - 2

### Інформація:

1. Інкапсуляція, успадкування, поліморфізм.
2. Модифікатори доступу. Особливість – модифікатор `internal`.
3. Успадкування – базовий синтаксис. Ключове слово `sealed`.
4. Виклик конструктору базового класу.
5. Реалізація поліморфізму в С#. Ключові слова `virtual`, `override`. `Sealed methods`.

### Джерела:

1. Троелсен-6, 201-207, 227-231, 233-237, 241-250
2. Ще раз [https://mva.microsoft.com/ru/training-courses/-c-1-8669?l=4SGxGjG1\\_6304984382](https://mva.microsoft.com/ru/training-courses/-c-1-8669?l=4SGxGjG1_6304984382)
3. [https://mva.microsoft.com/ru/training-courses/-c-1-8669?l=hFWxIjG1\\_3604984382](https://mva.microsoft.com/ru/training-courses/-c-1-8669?l=hFWxIjG1_3604984382)

### Завдання:

1. Переробити застосування з попереднього завдання з використанням успадкування та поліморфізму:
  - a. Створити базовий клас «летючий об'єкт», від якого успадкувати два або більше класів, наприклад «м'яч», «трикутник», «квадрат» - творчість вітається.
  - b. Для кожного класу реалізувати функцію «малювання в консолі» - наприклад, виглядати може так:

```
 . * * . . . . . * . . . . . * * * *
 * . . * . . . . * * . . . . * * * * *
 . * * . . . . * * * * * . . . . *
```
  - c. Для кожного класу реалізувати функції руху та відбивання (як варіант, використати той самий радіус обмежуючого кола, що і раніше)
  - d. Все, що спільне – винести в базовий клас, метод «намалюватись» зробити віртуальним (можливо, ще деякі з методів буде зручно зробити віртуальними).
  - e. На початку програми повинен створюватись масив типу базового класу, ініціалізуватись об'єктами кількох конкретних класів, і в результаті все це повинно літати по консолі, відбиватись від країв та одне від одного.
  - f. Важливо спробувати максимально витримувати принцип DRY – відсутності дублювання коду та логіки.

## Корисні посилання

1. Троелсен, Э. (2013). *Язык программирования C# 5.0 и платформа .NET 4.5, 6-е издание*. (вище згадується як Троелсен-б)  
<https://cobooks.tk/ru/book/78-endru-troelsen-c-50-i-platforma-net-45>
2. <https://mva.microsoft.com/ru/training-courses/-c-1-8669>  
On-line курс від співробітника Microsoft Ukraine Сергія Байдачного «Язык программирования C# (часть 1)»
3. <https://docs.microsoft.com/en-us/dotnet/csharp/index>  
«C# Guide» від Майкрософт – повна та найактуальніша документація з набором навчальних матеріалів. Будьте обережні з версіями та відмінностями між .Net 4.x та .Net Core!
4. <http://www.monodevelop.com/>  
Варіант середовища розробки для C# для не-windows систем.