

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ЗАХІДНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ФАКУЛЬТЕТ КОМП'ЮТЕРНИХ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

ЗАТВЕРДЖУЮ
Декан факультету комп'ютерних
інформаційних технологій
Ігор ЯКИМЕНКО
" 30 " _____ 2024 р.

ЗАТВЕРДЖУЮ
Проректор
з науково-педагогічної роботи
Віктор ОСТРОВЕРХОВ
" _____ 2024 р.

ЗАТВЕРДЖУЮ
Директор навчально-наукового
інституту новітніх освітніх
технологій
Святослав ПИТЕЛЬ
" 30 " _____ 2024 р.

РОБОЧА ПРОГРАМА

з дисципліни «Об'єктно-орієнтоване програмування»

Ступінь вищої освіти: бакалавр
Галузь знань – 12 «Інформаційні технології»
Спеціальність – 124 «Системний аналіз»
Освітньо-професійна програма – «Системний аналіз»

Кафедра комп'ютерних наук

Форма навчання	Курс	Семест р	Лекції (год.)	Лаб. (год.)	ІРС (год.)	Тренінг (год.)	СРС (год.)	Разом (год.)	Екз. (сем.)
Денна	2	3	30	16	3	6	65	120	3
Заочна	2	3	8	4	-	-	108	120	3

30.06.2024

Тернопіль – ЗУНУ
2024

Робоча програма складена на основі освітньо-професійної програми підготовки бакалаврів галузі знань 12 «Інформаційні технології» спеціальності 124 Системний аналіз, затвердженої Вченою радою ЗУНУ №10 від 23 червня 2023 р.

Робоча програма розроблена доцентом кафедри комп'ютерних наук, к.т.н. Володимиром МАНЖУЛОЮ

Робоча програма затверджена на засіданні кафедри комп'ютерних наук, протокол №1 від 27 серпня 2024 р.

Завідувач кафедри д.т.н, професор _____ Андрій Пукас



Розглянуто та схвалено групою забезпечення спеціальності 124 Системний аналіз, протокол №1 від 30 серпня 2024 р.

Голова групи
забезпечення спеціальності,
д.т.н, професор



Роман ПАСІЧНИК

Гарант ОП
д.т.н, професор



Роман ПАСІЧНИК

СТРУКТУРА РОБОЧОЇ ПРОГРАМИ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

“Об’єктно-орієнтоване програмування”

1. Опис дисципліни “Об’єктно-орієнтоване програмування”

Дисципліна – «Об’єктно-орієнтоване програмування»	Галузь знань, напрям підготовки, ступінь вищої освіти	Характеристика навчальної дисципліни
Кількість кредитів – 4	Галузь знань: 12 Інформаційні технології	Статус дисципліни: нормативна Мова навчання: Українська
Кількість залікових модулів: 5.	Спеціальність: 124 Системний аналіз	Рік підготовки: Денна – 2. Заочна – 2. Семестр: Денна – 3. Заочна – 3.
Кількість змістових модулів – 3	Ступінь вищої освіти: бакалавр	Лекції: Денна – 30 год. Заочна – 8 год. Лабораторні заняття: Денна – 16 год. Заочна – 4 год.
Загальна кількість годин – 120		Самостійна робота: Денна – 65 год. Тренінг – 6 Заочна – 108 год. Індивідуальна робота: Денна – 3 год.
Тижневих годин: 8 год., з них аудиторних: 3 год.		Вид підсумкового контролю: Екзамен

2. Мета й завдання вивчення дисципліни «Об’єктно-орієнтоване програмування»

2.1. Мета вивчення дисципліни

Мета викладання дисципліни “Об’єктно-орієнтоване програмування” полягає у наданні теоретичних знань та практичних навиків програмування з використанням об’єктно-орієнтованих мов програмування.

Мета проведення лекцій полягає у тому, щоб ознайомити студентів із головними методологічними та методичними питаннями об’єктно-орієнтованої парадигми програмування, та інструментарієм для створення об’єктно-орієнтованих програм від моделювання взаємозв’язків між об’єктами до створення програмного коду засобами об’єктно-орієнтованих мов програмування C#.

Мета проведення лабораторних занять полягає у тому, щоб виробити у студентів практичні навички побудови об’єктно-орієнтованих моделей та їх програмної реалізації засобами об’єктно-орієнтованої мов програмування C#.

2.2. Завдання вивчення дисципліни

Завданням дисципліни є набуття студентами теоретичних знань та практичних навичок принципів об'єктно-орієнтованої декомпозиції складних об'єктів, розробки проектів та написання на їх основі ефективних програм з використанням всіх переваг об'єктно-орієнтованої парадигми програмування.

У результаті вивчення дисципліни студенти повинні знати:

- фундаментальні концепції ООП;
- принципи програмування та реалізації ООП.
- мову об'єктно-орієнтованого програмування C#;
- перевантаження операторів;
- наслідування;
- поліморфізм
- шаблони;
- методи обробка виняткових ситуацій;
- ієрархію класів;

У результаті вивчення дисципліни студенти повинні вміти:

– визначати класи та об'єкти при застосуванні об'єктно-орієнтованих технологій програмування;

– здійснювати об'єктно-орієнтовану декомпозицію предметної області в умовах програмування складних об'єктів і систем за допомогою процедур об'єктно-орієнтованого аналізу, об'єктно-орієнтованого програмування, використовуючи визначення класів, поведінки об'єктів, структури даних та їх взаємозв'язки;

– розробляти об'єктно-орієнтовану модель предметної галузі за допомогою мов об'єктно-орієнтованого моделювання в умовах програмування складних об'єктів та систем, використовуючи об'єктно-орієнтовану нотацію складних систем.

2.3. Найменування та опис компетентностей, формування котрих забезпечує вивчення дисципліни:

ФК11. Здатність реалізовувати фази та ітерації життєвого циклу програмних систем та інформаційних технологій на основі відповідних моделей і підходів розробки програмного забезпечення.

ФК13. Здатність обґрунтовано обирати та освоювати інструментарій з розробки та супроводження програмного забезпечення

2.4. Передумови для вивчення дисципліни.

Дисципліни, які повинні бути вивчені попередньо:

- «Основи програмування»;

2.5. Результати навчання

У результаті вивчення курсу «Об'єктно-орієнтоване програмування» студенти повинні:

РН03. Знати основні процеси, фази та ітерації життєвого циклу програмного забезпечення.

РН06. Уміння вибирати та використовувати відповідну задачі методологію створення програмного забезпечення.

PH14. Застосовувати на практиці інструментальні програмні засоби доменного аналізу, проектування, тестування, візуалізації, вимірювань та документування програмного забезпечення.

PH15. Мотивовано обирати мови програмування та технології розробки для розв'язання завдань створення і супроводження програмного забезпечення.

PH18. Знати та вміти застосовувати інформаційні технології обробки, зберігання та передачі даних.

PH22. Знати та вміти застосовувати методи та засоби управління проектами.

3. Зміст дисципліни “Об’єктно-орієнтоване програмування”

Змістовий модуль 1. Основи C # і платформа .NET.

Тема 1. Мова C # і платформа .NET

Створення C #. Зв'язок C # з середовищем .NET Framework. Загальномовне виконуюча середу CLR. Загальна система типів CTS. Загальномовне специфікація CLS. Збірки. Проміжний мова CIL. Класи і простору імен. Загальномовна інфраструктура CLI.

Література: 1,3,4.

Тема 2. Основи C #

Проста програма на C #. Коментарі. Змінні. Область видимості змінних, константи. Типи даних. Літерали. Перетворення типів. Арифметичні оператори. Оператори відносини і логічні оператори. Оператор присвоювання. Порозрядні оператори. Тернарний оператор. Умовні оператори. Циклічні оператори: for, while, do while і foreach. Оператори переходу.

Література: 2,3,4.

Тема 3. Масиви, кортежі і рядки

Масиви. Багатовимірні масиви. Ступінчасті (зубчасті) масиви масивів. Клас Array. Масиви в якості параметрів. Кортежі. Рядки. Клас StringBuilder. Форматуючі рядки. Регулярні вирази.

Література: 2,3,4.

Змістовий модуль 2. Реалізація принципів ООП мовою C#.

Тема 4. Класи і об'єкти в C#.

Класи. Клас Object. Створення об'єктів. Методи. Конструктори. Збірка сміття і деструктори. Ключове слово this. Доступ до членів класу. Модифікатори параметрів. Необов'язкові та іменовані аргументи. Рекурсія. Ключове слово static. Індексатори. Властивості. Модифікатори доступу в аксесорах.

Література: 2,3,4,5,6.

Тема 5. Перевантаження методів класу

Перевантаження методів. Перевантаження конструкторів. Перевантаження індиксаторів. Основи перевантаження операторів. Перевантаження операторів відношення і операторів true – false. Перевантаження логічних операторів. Оператори перетворення.

Література: 2,3,4,5,6.

Тема 6. Успадкування та поліморфізм

Основи успадкування. Захищений доступ і виключення успадкування. Конструктори та успадкування. Успадкування та приховування імен. Посилання на базовий клас і об'єкти похідних класів. Віртуальні методи, властивості і індексатори. Абстрактні класи.

Література: 2,3,4,5,6.

Тема 7. Інтерфейси, структури та перерахування

Інтерфейси. Інтерфейсні посилання. Інтерфейсні властивості і індексатори. Спадкування інтерфейсів. Явна реалізація інтерфейсу. Структури. Перерахування.

Література: 2,3,4,5,6.

Змістовий модуль 3. Програмування обробки винятків та подій в C#.

Тема 8. Обробка винятків в C#

Основи обробки винятків. Приховати винятків. Клас Exception. Конфігурування стану виключення. Винятки рівня системи та програми. Обробка численних винятків. Оператори throw і finally. Винятки, пов'язані з пошкодженим станом (Corrupted State Exceptions). Ключові слова checked і unchecked.

Література: 7,8,9,10.

Тема 9. Делегати та лямбда-вирази

Делегати. Груповий виклик і адресація делегованих методів. Коваріантність і контраваріантних делегатів. Делегати Action <T> і Func <T>. Анонімні методи. Лямбда-вирази.

Література: 7,8,9,10.

Тема 10. Події

Події. Аксесор подій. Обробка подій в середовищі .NET Framework.

Література: 7,8,9,10.

Тема 11. Узагальнення

Огляд узагальнень. Узагальнені класи. Обмежені типи. Обмежені класи. Обмежені інтерфейси і конструктори. Обмеження посилального типу і типу значення. Ієрархії узагальнених класів. Засоби узагальнень. Узагальнені методи. Узагальнені структури. Узагальнені делегати. Узагальнені інтерфейси. Модифікація узагальнених методів. Коваріантність і контраваріантних в узагальненнях.

Література: 7,8,9,10.

4. Структура залікового кредиту з дисципліни “Об’єктно-орієнтоване програмування”

(Денна форма навчання)

	Кількість годин					
	Лекції	Лабораторні заняття	Самостійна робота	ІРС	Тренінг	Контрольні заходи
Змістовий модуль 1. Основи C # і платформа .NET.						
Тема 1. Мова C # і платформа .NET	2		6	1	2	Усне опитування та тестування
Тема 2. Основи C #	2	2	6			Усне опитування та тестування
Тема 3. Масиви, кортежі і рядки	4	2	6			Усне опитування та тестування
Змістовий модуль 2. Реалізація принципів ООП мовою C#.						
Тема 4. Класи і об'єкти в C#	2	1	6	1	2	Усне опитування та тестування
Тема 5. Перевантаження методів класу	2	1	6			Усне опитування та тестування
Тема 6. Успадкування та поліморфізм	4	2	6			Усне опитування та тестування
Тема 7. Інтерфейси, структури та перерахування	4	1	6			Усне опитування та тестування
Змістовий модуль 3. Програмування обробки винятків та подій в C#.						
Тема 8. Обробка винятків в C#	2	2	6	1	2	Усне опитування та тестування
Тема 9. Делегати та лямбда-вирази	2	2	6			Усне опитування та тестування
Тема 10. Події	2	1	6			Усне опитування та тестування
Тема 11. Узагальнення	4	2	5			Усне опитування та тестування
Разом	30	16	65	3	6	

(Заочна форма навчання)

	Кількість годин		
	Лекції	Лабораторні заняття	Самостійна робота
Змістовий модуль 1. Основи C # і платформа .NET.			
Тема 1. Мова C # і платформа .NET	1		10
Тема 2. Основи C #	1		10
Тема 3. Масиви, кортежі і рядки	1	1	10
Змістовий модуль 2. Реалізація принципів ООП мовою C#.			
Тема 4. Класи і об'єкти в C#	1	1	10
Тема 5. Перевантаження методів класу			10
Тема 6. Успадкування та поліморфізм	1	1	10
Тема 7. Інтерфейси, структури та перерахування	1	1	8
Змістовий модуль 3. Програмування обробки винятків та подій в C#.			
Тема 8. Обробка винятків в C#	1		10
Тема 9. Делегати та лямбда-вирази	1		10
Тема 10. Події			10
Тема 11. Узагальнення			10
Разом	8	4	108

5. Тематика лабораторних занять

Лабораторна робота № 1

Тема: Основи C# та .NET.

Мета: Навчитися створювати консольні додатки мовою C# на платформі .NET.

Література: 1, 2, 3, 4.

Лабораторна робота № 2

Тема: Використання масивів, кортежів і рядків у C#.

Мета: Навчитися використовувати масиви, кортежі і рядки у C#.

Література: 2, 3, 4, 5.

Лабораторна робота № 3

Тема: Конструювання об'єктів класів у C#.

Мета: Засвоєння методики створення конкретних класів з використанням мови програмування C#.

Література: 4, 5, 8, 10.

Лабораторна робота № 4

Тема: Реалізація наслідування.

Мета: Засвоєння методики реалізації наслідування з використанням мови програмування C#.

Література: 2, 4, 10.

Лабораторна робота № 5

Тема: Поліморфізм.

Мета: Засвоєння методики реалізації поліморфізму з використанням мови програмування C#.

Література: 3, 4, 5, 6.

Лабораторна робота №6

Тема: Перевантаження методів на C#.

Мета: Засвоєння методики перевантаження унарних та бінарних операторів для класів корисувача.

Література: 2, 4, 5, 6.

Лабораторна робота № 7

Тема: Розробка інтерфейсів.

Мета: Навчитись використовувати можливості поліморфізму на основі інтерфейсів.

Література: 3, 4, 7.

Лабораторна робота № 8

Тема: Обробка виняткових ситуацій на C#.

Мета: Засвоєння методики відлагодження програм та обробки програм на етапі виконання засобами об'єктно-орієнтованого програмування на C#.

Література: 5, 6, 7.

Лабораторна робота № 9

Тема: Використання делегатів.

Мета: Засвоєння принципів роботи з делегатами на C#.

Література: 2, 8, 9, 10.

6. Самостійна робота

Для успішного вивчення і засвоєння дисципліни «Об'єктно-орієнтоване програмування» студенти повинні володіти значним обсягом інформації, частину якої вони отримують і опрацьовують шляхом самостійної роботи. Самостійна робота полягає в опрацюванні сучасної навчальної і наукової фахової літератури.

Результати самостійної роботи студента з дисципліни « Об'єктно-орієнтоване програмування» оцінюються за 100-бальною шкалою у вигляді аналітичного огляду наукових джерел або літератури відповідно до обраної з переліку теми.

- 1 Процес розробки програмного забезпечення, його мета, головні етапи.
- 2 Цілі програмування та розробки програмного забезпечення.
 1. Використання змінних.
 2. Загальноживані оператори.
 3. Умовні конструкції.
 4. Цикли. Типи циклів.
 5. Процедури.
 6. Обробка помилок.
 7. Скалярні стандартні типи даних в мові C#.

8. Скалярні типи користувача в мові C#.
9. Основні операції в мові C#.
10. Загальна структура програми мовою C#.
11. Умовні оператори мови C#.
12. Оператори циклу у мові C#.
13. Введення і виведення інформації у мові C#.
14. Методи в мові C#.
15. Рекурсія. Пряма та непряма рекурсія.
16. Типи параметрів у мові C#.
17. Масиви у мові C#.
18. Основні принципи об'єктно-орієнтованого підходу в програмуванні.
19. Поняття інкапсуляції. Організація взаємодії об'єктів.
20. Типи-значення та типи-посилання.
21. Способи створення об'єктів.
22. Способи знищення об'єктів. Збирач сміття.
23. Базові елементи програми. Як вони відображаються графічно.
24. Принцип компіляції у .NET.
25. Структура простої програми мовою C#.
26. Основні скалярні типи даних мови C#.
27. Правила іменування змінних.
28. Основні операції мови C#.
29. Оператори інкременту та декременту.
30. Явне і неявне перетворення типів.
31. Принцип обробки виключень в мові C#.
32. Клас в мові C#.
33. Методи класу. Звертання до методу класу.
34. Конструктори класу.
35. Деструктор класу.
36. Утилізація об'єктів, що не використовуються програмою.
37. Масиви в мові C.
38. Цикл foreach.
39. Модифікатори параметрів методів класу.
40. Структури в мові C#.
41. Колекції в мові C#.
42. Перезавантажені методи класу.
43. Зв'язок між типом та екземпляром масива.
44. Основні характеристики об'єкта: ідентичність, стан і поведінку.
45. Різниця між типами-значеннями та типами-посиланнями.
46. Основні простори імен у .NET.
47. Особливості конструктора за замовчуванням.
48. Роль збирача сміття.

7. Тренінг з дисципліни

Тема: реалізація консольного програмного додатку.

Порядок проведення:

- здійснити об'єктно-орієнтований аналіз предметної області;

- реалізувати об'єктно-орієнтовану модель на основі методів об'єктно-орієнтоване проектування;
- реалізувати додаток на мові об'єктно-орієнтованого програмування;
- провести тестування програми;
- представити результати роботи у вигляді короткого звіту: діаграми варіантів використання та класів, реалізація, результати тестування.

Література: 2, 10.

Варіанти завдань для тренінгу з дисципліни "Об'єктно-орієнтоване програмування"

1. Реалізація об'єктно-орієнтованої програми для імітації роботи ліфта.
2. Реалізація об'єктно-орієнтованої програми для імітації роботи музичного центру.
3. Реалізація об'єктно-орієнтованої програми для імітації роботи холодильника.
4. Реалізація об'єктно-орієнтованої програми для імітації роботи пральної машина.
5. Реалізація об'єктно-орієнтованої програми для імітації роботи охоронної системи приміщення (в тому числі - протипожежна).
6. Реалізація об'єктно-орієнтованої програми для імітації роботи телефонного апарату.
7. Реалізація об'єктно-орієнтованої програми для імітації роботи системи забезпечення безпеки руху.
8. Реалізація об'єктно-орієнтованої програми для імітації роботи автоматичної системи контролю стану систем літака.
9. Реалізація об'єктно-орієнтованої програми для імітації роботи автоматичної системи контролю стану систем автомобіля.
10. Реалізація об'єктно-орієнтованої програми для імітації роботи банкомата.
11. Реалізація об'єктно-орієнтованої програми для імітації роботи автомата по продажу цукерок.
12. Реалізація об'єктно-орієнтованої програми для імітації роботи телевізора.
13. Реалізація об'єктно-орієнтованої програми для імітації роботи системи опрацювання текстів.
14. Реалізація об'єктно-орієнтованої програми для імітації роботи мікрохвильової печі.
15. Реалізація об'єктно-орієнтованої програми для імітації роботи турнікета метрополітену.
16. Реалізація об'єктно-орієнтованої програми для імітації процесу вирощування рослини.
17. Реалізація об'єктно-орієнтованої програми для імітації роботи системи зрошування газону.
18. Реалізація об'єктно-орієнтованої програми для імітації роботи системи кухонного комбайну.

19. Реалізація об'єктно-орієнтованої програми для імітації роботи системи багатофункціонального дистанційного пульта управління (tv, cd, vcd, sat,...).

20. Реалізація об'єктно-орієнтованої програми для імітації роботи тестера (амперметр, омметр, вольтметр).

21. Реалізація об'єктно-орієнтованої програми для імітації роботи інженерного калькулятора (dec, hex, bin, ...).

22. Реалізація об'єктно-орієнтованої програми для імітації роботи системи відеоспостереження.

23. Реалізація об'єктно-орієнтованої програми для імітації роботи системи опалення приміщень (газ. котел, насос, ...).

24. Реалізація об'єктно-орієнтованої програми для опису функціонування факультету ВНЗ.

8. Засоби оцінювання та методи демонстрування результатів.

В процесі вивчення дисципліни “Об'єктно-орієнтоване програмування” використовуються наступні методи оцінювання навчальної роботи студента:

- поточне опитування;
- стандартизовані тести;
- залікове модульне тестування та опитування;
- презентації результатів виконаних завдань та досліджень;
- студентські презентації та виступи на наукових заходах;
- тренінги;
- інші види індивідуальних та групових завдань.

9. Критерії, форми поточного та підсумкового контролю

Підсумковий бал (за 100-бальною шкалою) з дисципліни “Об'єктно-орієнтоване програмування” визначається як середньозважена величина, в залежності від питомої ваги кожної складової залікового кредиту:

Модуль 1		Модуль 2		Модуль 3	Модуль 4	Модуль 5
10%	10 %	10 %	10 %	5%	15%	40%
Поточне оцінювання	Модульний контроль 1	Поточне оцінювання	Модульний контроль 2	Тренінги	Самостійна робота	Екзамен
Оцінювання виконаних лабораторних робіт: № 1 -8 по 10 балів кожна = 80 балів; Модульна робота – 20 балів.	Підсумкове тестування за темами №1-10 – 25 питань по 4 бали = 100 балів	Оцінювання виконаних лабораторних робіт: № 9 -18 по 10 балів кожна = 100 балів.	Підсумкове тестування за темами №11-24 – 25 питань по 4 бали.	Оцінювання розробленого проекту – 100 балів.	Оцінювання аналітичного огляду наукових джерел або літератури відповідно до обраної теми – 100 балів.	1. Тестові завдання (30 тестів по 2 бали за тест) – 60 балів 2. Завдання 1 – 20 балів 3. Завдання 2 – 20 Балів

Шкала оцінювання:

За шкалою Університету	За національною шкалою	За шкалою ECTS
90-100	відмінно	A (відмінно)
85-89	добре	B (дуже добре)
75-84		C (добре)
65-74	задовільно	D (задовільно)
60-64		E (достатньо)
35-59	незадовільно	FX (незадовільно з можливістю повторного складання)
1-34		F (незадовільно з обов'язковим повторним курсом)

10. Інструменти, обладнання та програмне забезпечення, використання яких передбачає навчальна дисципліна

№	Найменування	Номер теми
1.	Мультимедійний проектор	1-24
2.	Проекційний екран	1-24
3.	Комунікаційне програмне забезпечення (Internet Explorer, Google Chrome, Firefox)	1-24
4.	Операційна система Windows, наявність доступу до мережі Internet	1-24
5.	Персональні комп'ютери	1-24
6.	Комунікаційне програмне забезпечення (Zoom) для проведення занять у режимі онлайн (за необхідності)	1-24
7.	Комунікаційна навчальна платформа (Moodle) для організації дистанційного навчання (за необхідності)	1-24
8.	Базове програмне забезпечення Microsoft Office	1-24
9.	Спеціалізоване програмне забезпечення: - Середовище розробки Microsoft Visual Studio	3-24

Рекомендовані джерела інформації

1. <https://msdn.microsoft.com/> – керівництво по продуктах Microsoft
2. International Standard ISO/IEC 14882:2014(E) – Programming Language C++, ISBN13: 978-0321563842: [Електронний ресурс]. – Режим доступу: <https://isocpp.org/std/thestandard>.
3. Josph Albahari. C# 8.0 Pocket Reference. – 1st edition, O'Reilly LCF Publishing. C#. – 1st edition, Learn Coding Fast
4. Mark J. Price C# 8.0 and .NET Core 3.0. – 4th edition. Packt Publishing

5. Andrew Stellman. Head First C#. – 4th edition, O'Reilly Media, Inc. – 2020.
6. Mikael Olsson. C# 10 Quick Syntax Reference. – 4th Ed. Apress. – 2023
7. Joe Mayo. C# Cookbook: Modern Recipes for Professional Developers. 1st Ed.
8. C# OOP (Object-Oriented Programming) - W3Schools. [Электронный ресурс]. – Режим доступа: https://www.w3schools.com/cs/cs_oop.php
9. Learn C#. Free courses, tutorials, videos, and more about learning the programming language C#. [Электронный ресурс]. – Режим доступа: <https://dotnet.microsoft.com/en-us/learn/csharp>
10. Object-Oriented programming (C#). [Электронный ресурс]. – Режим доступа: <https://learn.microsoft.com/en-us/dotnet/csharp/fundamentals/tutorials/oop>.