Чернівецький національний університет імені Юрія Федьковича
(повне найменування закладу вищої освіти)

Факультет математики та інформатики
(назва факультету/навчально-наукового інституту)

Кафедра математичного моделювання
(назва кафедри, що забезпечує викладання)

"ЗАТВЕРДЖУЮ"
Декан факультету
математики та інформатики
_____ Ольга МАРТИНЮК
"28" серпня 2026 року

РОБОЧА ПРОГРАМА
навчальної дисципліни
Платформи корпоративних інформаційних систем
(назва навчальної дисципліни)

обов'язкова
(вказати: обов'язкова)

Освітньо-професійна програма Інформаційні технології та управління проєктами
(назва програми)

Спеціальність 122 Комп'ютерні науки
(вказати: код, назва)

Галузь знань 12 Інформаційні технології
(вказати: шифр, назва)

Рівень вищої освіти перший (бакалаврський)
(вказати: перший (бакалаврський) / другий (магістерський) / третій (освітньо-науковий)

Факультет математики та інформатики
(назва факультету/ навчально-наукового інституту, на якому здійснюється підготовка фахівців за вказаною освітньою програмою)

Мова навчання українська
(вказати: на якій мові читається дисципліна)

Чернівці 2026 рік

Робоча програма навчальної дисципліни «*Платформи корпоративних інформаційних систем*» складена відповідно до освітньо-професійної програми «Інформаційні технології та управління проектами»

Розробник:
Горбатенко Микола Юрійович, доцент кафедри математичного моделювання, кандидат фіз.-мат. наук, доцент

Викладач, що забезпечує читання даної навчальної дисципліни:
Горбатенко Микола Юрійович, доцент кафедри математичного моделювання, кандидат фіз.-мат. наук, доцент

Погоджено з гарантом ОП _____ **Ігор ЮРЧЕНКО**

Затверджено на засіданні кафедри математичного моделювання
Протокол № 1 від «27» серпня 2026 року

Завідувач кафедри _____ **Ігор ЧЕРЕВКО**

Схвалено методичною радою факультету математики та інформатики
Протокол № 1 від «28» серпня 2026 року

Голова методичної ради _____ **Тоня ФРАТАВЧАН**

Мета навчальної дисципліни: формування навичок проектування, розробки, встановлення додатків, використання сучасних методів програмування та сучасних середовищ розробки. Переваги навчальної дисципліни:

- вивчення сучасних стандартів та підходів до розробки;
- розгляд розробки актуальних типів програмних додатків;
- застосування ефективних практик для написання швидких, стабільних та безпечних додатків;
- коректні підходи при роботі в стандартних та нештатних ситуаціях;
- ефективні принципи організації взаємодії залежностей в середині додатку та коректна робота з зовнішніми залежностями.

Результати навчання: у результаті вивчення навчальної дисципліни студент повинен

знати: теоретичні положення про .Net, теоретичні положення про C#, теоретичні положення про клієнтську обробку, теоретичні положення про взаємодію з користувачем, теоретичні положення про обробку виключних ситуацій, теоретичні положення про створення багатопоточних додатків.

вміти: створювати компоненти .Net, за допомогою мови C#, створювати та супроводжувати проекти, коректно обробляти нештатні ситуації, вміти користуватися основними можливостями IDE Visual Studio, Visual Studio Code.

Студент повинен оволодіти програмним матеріалом, виконати лабораторні роботи, засвоїти теоретичний матеріал.

Під час вивчення дисципліни, відповідно до освітньо-професійної програми, формуються наступні

загальні компетентності:

ЗК1. Здатність до абстрактного мислення, аналізу та синтезу.

ЗК2. Здатність застосовувати знання у практичних ситуаціях.

ЗК3. Знання та розуміння предметної області та розуміння професійної діяльності.

ЗК6. Здатність вчитися й оволодівати сучасними знаннями.

ЗК7. Здатність до пошуку, оброблення та аналізу інформації з різних джерел.

ЗК8. Здатність генерувати нові ідеї (креативність).

ЗК10. Здатність бути критичним і самокритичним.

ЗК11. Здатність приймати обґрунтовані рішення.

фахові компетентності:

ФК8. Здатність проєктувати та розробляти програмне забезпечення із застосуванням різних парадигм програмування: узагальненого, об’єктно-орієнтованого, функціонального, логічного, з відповідними моделями, методами й алгоритмами обчислень, структурами даних і механізмами управління.

ФК10. Здатність застосовувати методології, технології та інструментальні засоби для управління процесами життєвого циклу інформаційних і програмних систем, продуктів і сервісів інформаційних технологій відповідно до вимог замовника.

ФК12. Здатність забезпечити організацію обчислювальних процесів в інформаційних системах різного призначення з урахуванням архітектури, конфігурування, показників результативності функціонування операційних систем і системного програмного забезпечення.

та отримуються наступні **програмні результати навчання**:

ПРН9. Розробляти програмні моделі предметних середовищ, вибирати парадигму програмування з позицій зручності та якості застосування для реалізації методів та алгоритмів розв'язання задач в галузі комп'ютерних наук.

ПРН10. Використовувати інструментальні засоби розробки клієнтсерверних застосувань, проєктувати концептуальні, логічні та фізичні моделі баз даних, розробляти та оптимізувати запити до них, створювати розподілені бази даних, сховища та вітрини даних, бази знань, у тому числі на хмарних сервісах, із застосуванням мов веб-програмування.

Опис навчальної дисципліни

Загальна інформація

Форма навчання	**Рік підготовки**	**Семестр**	**Кількість**		**Кількість годин**						**Вид підсумкового контролю**
			кредитів	**годин**	**лекції**	**практичні**	**семінарські**	**лабораторні**	**самостійна робота**	**індивідуальні завдання**	
Денна	3,4	6,7	8	240	60	-	-	60	120	-	Залік, Іспит
		6	4	120	30	-	-	30	60	-	Залік
		7	4	120	30	-	-	30	60	-	Іспит

Структура змісту навчальної дисципліни

6 семестр

Назви змістових модулів і тем навчальних занять	Кількість годин											
	денна форма						заочна форма					
	усього	у тому числі					усього	у тому числі				
		лекц.	прак./ семін.	лаб.	інд.	с.р.		лекц.	прак./ сем.	лаб.	інд.	с.р.
1	2	3	4	5	6	7	8	9	10	11	12	13
Теми навчальних занять	**Змістовий модуль 1. Мова C#**											
Тема 1. Вступ. Огляд .Net екосистеми. Основи C#.	16	4	-	4	-	8	-	-	-	-	-	-

Тема 2. Основні конструкції, виняткові ситуації, функції, тестування та налагоджування.	16	4	-	4	-	8	-	-	-	-	-	-
Тема 3. Створення власних типів та стандарти.	18	4	-	4	-	10	-	-	-	-	-	-
Тема 4. Реалізація інтерфейсів, делегатів та події.	18	4	-	4	-	10	-	-	-	-	-	-
Разом за ЗМ1	68	16	-	16	-	36	-	-	-	-	-	-
Теми навчальних занять	**Змістовий модуль 2. .Net та середовище виконання (Runtime)**											
Тема 5. Ефективна робота з ресурсами, інтернаціоналізація, регулярні вирази, серіалізація.	14	4	-	4	-	6	-	-	-	-	-	-
Тема 6. LINQ.	15	4	-	5	-	6	-	-	-	-	-	-
Тема 7. Задачі (Task) та паралельне програмування	17	4	-	5	-	8	-	-	-	-	-	-
Тема 8. Типи додатків та їх використання в реальному житті.	6	2	-	-	-	4	-	-	-	-	-	-
Разом за ЗМ 2	52	14	-	14	-	24	-	-	-	-	-	-
Усього годин	120	30	-	30	-	60	-	-	-	-	-	-

7 семестр

Назви змістових модулів і тем навчальних занять	Кількість годин											
	денна форма						Заочна форма					
	усього	у тому числі					усього	у тому числі				
		лекц.	прак./ семін.	лаб.	інд.	с.р.		лекц.	прак./ семін.	лаб.	інд.	с.р.
1	2	3	4	5	6	7	8	9	10	11	12	13
Теми навчальних занять	**Змістовий модуль 1. Дані та веб-додатки**											
Тема 1. ADO.NET.	16	4	-	4	-	8	-	-	-	-	-	-
Тема 2. Entity Framework Core.	16	4	-	4	-	8	-	-	-	-	-	-
Тема 3. ASP.NET MVC та Razor pages.	13	3	-	3	-	7	-	-	-	-	-	-
Тема 4. ASP.NET Web API.	13	3	-	3	-	7	-	-	-	-	-	-
Разом за ЗМ1	58	14	-	14	-	30	-	-	-	-	-	-

Теми навчальних занять	Змістовий модуль 2. Сервіси та клієнтська частина.											
Тема 5. Blazor.	11	2	-	3	-	6	-	-	-	-	-	-
Тема 6. Використання спеціалізованих сервісів.	14	4	-	4	-	6	-	-	-	-	-	-
Тема 7. MAUI та захист даних в додатках.	14	4	-	4	-	6	-	-	-	-	-	-
Тема 8. Typescript та взаємодія з фронт-енд додатками.	15	4	-	5	-	6	-	-	-	-	-	-
Тема 9. Типові архітектури, методи розгортання та верифікація.	8	2	-	-	-	6	-	-	-	-	-	-
Разом за ЗМ 2	62	16	-	16	-	30	-	-	-	-	-	-
Усього годин	120	30	-	30	-	60	-	-	-	-	-	-

Тематика лекційних занять з переліком питань

6 семестр

1	Вступ. Огляд .Net екосистеми. Основи C#. Створення програми з та без використання функції Main. Інтегровані середовища розробки та їх використання.
2	Основні конструкції, виняткові ситуації, функції, тестування та налагоджування. Використання операторів та операндів, створення власних функції. Типові підходи до обробки виняткових ситуацій та створення власних типів для їх обробки. Підходи для визначення проблемних частин коду та їх усунення.
3	Створення власних типів, стандарти оголошення та використання. Поля, методи, властивості, константи. Мутація значень та заборона мутації, принципи використання.
4	Реалізація інтерфейсів, делегатів та події. Поняття інтерфейсу, декларація функціоналу, реалізація функціоналу за замовчуванням, явне та неявне використання. Декларація та використання делегатів, їх типові приклади. Оголошення події, їх обробників та типові підходи до використання.
5	Ефективна робота з ресурсами, інтернаціоналізація, регулярні вирази, серіалізація. Типові практики для оптимізації використання пам’яті та вибір коректного типу або створення власного. Принципи розробки багатомовних додатків, підтримка часових поясів. Підходи до роботи з регулярними виразами. Типи серіалізаторів та їх використання.
6	LINQ. Принципи роботи з різними джерелами даних, вирази, використання методів та побудова за допомогою запитів, методи об’єднання джерел та використання для розв’язання типових задач.

7	Задачі (Task) та паралельне програмування. Поняття потоків, задач, типові проблеми та підходи при розв'язанні таких задач. Використання Task, TPL, Parallel, методи синхронізації та використання результатів, отриманих на етапах попередніх обчислень.
8	Потокозахищені колекції, їх використання. Типи додатків та їх використання в реальному житті. Розгляд типів додатків, які можна розробляти на платформі, коли потрібно використовувати такий тип додатку.

7 семестр

1	ADO.NET. Підходи до роботи з реляційними базами даних, підключення до бази даних, пул активних підключень та діагностика, робота з даними та об'єктами, типові практики для роботи з даними.
2	Entity Framework Core. Провайдери даних, робота з реляційними та нереляційними типами даних, підходи до зв'язування об'єктів та їх використання.
3	ASP.NET MVC та Razor pages. Приклади застосування додатків такого типу, вибір підходу, використання, основні блоки та залежності.
4	ASP.NET Web API. Розробка API, основні структурні блоки, побудова документації, OpenAPI, підтримувані формати серіалізації та розширення функціоналу, логування.
5	Blazor. Клієнтська та серверна частини, гібридні додатки, розробка компонентів, основних блоків та їх взаємодія, приклади.
6	Використання спеціалізованих сервісів. Приклади побудови сервісів GraphQL, gRPC, oData, SignalR, Azure Functions та їх застосування.
7	MAUI та захист даних в додатках. Розробка десктопних та мобільних додатків. Розгляд методів шифрування, захисту, хешування, верифікації даних.
8	Typescript та взаємодія з фронт-енд додатками. Розгляд переваг Typescript для розробки клієнтської частини, приклади застосування. Розгортання додатків.

Тематика лабораторних занять

6 семестр

№	Назва теми
Змістовий модуль 1. Мова C#	
1	Розробка додатку за допомогою базових конструкцій.
2	Написання тестів та методи налагоджування додатку.
3	Створення власних типів згідно стандартів.
4	Делегати та події.
Змістовий модуль 2. .Net та середовище виконання (Runtime)	
5	Інтернаціоналізація, серіалізація, оптимізація.
6	LINQ.
7	Розробка асинхронних додатків.
8	Використання потокозахищених колекцій.

7 семестр

№	Назва теми
Змістовий модуль 1. Дані та веб-додатки	
1	Створення додатку з використанням ADO.NET
2	Створення додатку з використанням Entity Framework Core.
3	Створення додатку з використанням Asp.Net MVC.
4	Створення додатку з використанням ASP.NET Web API.
Змістовий модуль 2. Сервіси та клієнтська частина.	
5	Створення додатку з використанням Blazor.
6	Створення додатку з використанням спеціалізованого сервісу.
7	Створення додатку з використанням MAUI.
8	Створення клієнт-серверного додатку з використанням Typescript. Аналіз методів розгортання.

Індивідуальні науково-дослідні завдання (ІНДЗ)

Студент може індивідуально виконувати додаткові завдання навчально-дослідницької спрямованості за завданнями, наданими викладачем.

Студенти можуть отримати до 10 балів в рахунок ІНДЗ, якщо самостійно зареєструються на безкоштовних релевантних курсах платформи Coursera (або подібних), пройдуть навчання, отримають відповідний сертифікат і надішлють його на сайт дистанційного навчання викладачу разом з детальним звітом з практичних завдань пройденого курсу (постановки задач, коди виконаних програм, пояснення коду) та скріншотом успішності на курсі. Кількість балів буде виставлена пропорційно до навчальних результатів студента згідно зі статистикою платформи.

* ІНДЗ – для змістового модуля, або в цілому для навчальної дисципліни визначається викладачем, з урахуванням специфіки дисципліни.

Завдання для самостійної роботи студентів

Самостійна робота складається з повторення матеріалу, засвоєного на лекціях, самостійного опанування частини теоретичного матеріалу, роботи з контрольними запитаннями та завданнями.

Студенти можуть отримувати 1 бал в рахунок виконання завдань СРС під час кожного лекційного заняття за правильні відповіді на запитання лектора, активне обговорення багатоваріантних підходів до рішення представленої лектором проблеми (для активізації пошукової та дослідної діяльності студентів).

6 семестр

№	Назва теми	Завдання для самостійної роботи	К-сть год.
1	Теми 1-8	підготовка до лекційних занять	20
2	Теми 1-8	підготовка до лабораторних занять	30
3	Теми 1-8	підготовка до підсумкового модуль-контролю	10

7 семестр

№	Назва теми	Завдання для самостійної роботи	К-сть год.
1	Теми 1-8	підготовка до лекційних занять	20
2	Теми 1-8	підготовка до лабораторних занять	30
3	Теми 1-8	підготовка до підсумкового модуль-контролю	10

Методи навчання

Використовуються як класичні методи навчання, такі як лекційні та лабораторні заняття, так і Google Classroom, де розміщуються матеріали для ознайомлення та завдання для лабораторних робіт з можливістю завантаження отриманих результатів під час самостійної роботи та роботи над лабораторними роботами, а також перевагу надається використання Google Forms для проведення підсумкового оцінювання за наявності такої можливості. Google Meet використовується за умови змішаної або дистанційної форми навчального процесу.

Система контролю та оцінювання

Критерієм підсумкового оцінювання є досягнення студентом мінімальних порогових рівнів оцінок (балів) за кожним передбаченим результатом навчання.

Система оцінювання рівня навчальних досягнень ґрунтується на принципах ECTS та є накопичувальною. Протягом семестру студенти виконують 8 лабораторних робіт. Кожна лабораторна робота оцінюється в 10 балів.

Виконуючи завдання лабораторної роботи, студент повинен оформити і завантажити для подальшої перевірки на сайт електронного навчання звіт разом із працездатними файлами програмної реалізації завдання.

50% балів, відведених на оцінювання ЛР, студент отримує за працюючий програмний продукт, в якому реалізовано всі завдання ЛР та оформлений звіт. Решта 50% балів виставляється після захисту студентом виконаного звіту. На захисті звіту з лабораторної роботи студент має відповісти на питання щодо постановки задачі та розробленої реалізації. При відповіді на теоретичні питання та питання щодо програмної реалізації у випадку неістотної помилки знімається 10-20% балів, а у випадку істотної 20-40% балів, якщо ж студент не опанував теоретичний матеріал, плутається в означеннях, наводить невірні твердження, то знімається до 50% балів від усієї суми балів за лабораторну роботу.

Максимальна кількість, яку можна набрати на підсумковому модулі (тестування) – 20 балів в 6-му та 7 семестрах.

Підсумкова оцінка виставляється за результатами суми балів, набраних на змістових модулях під час семестру та підсумковому модулі згідно з нижченаведеною таблицею.

Методи формування професійної компетентності: розповідь, пояснення, бесіда, демонстрація, візуалізація, дискусія тощо. Методи формування практичних умінь та навичок: розв’язування задач лабораторних робіт,

виконання завдань, розробка та аналіз алгоритмів і програмного коду, захист звітів з лабораторних робіт.

Засобами оцінювання та демонстрування результатів навчання є: стандартизовані тести; аналітичні звіти з лабораторних робіт; презентації результатів виконаних завдань та досліджень ІНДЗ, усний контроль у вигляді індивідуального та фронтального опитування на лекціях та лабораторних заняттях.

Формами поточного контролю є усна чи письмова (тестування, лабораторна робота, ІНДЗ) відповідь студента.

Формою підсумкового контролю є залік в 6-му семестрі та іспит в 7-му семестрі.

Критерії оцінювання поточного та підсумкового контролю

6 семестр

<table>
<tr><td colspan="8">Поточне оцінювання (аудиторна та самостійна робота)</td><td>Кількість балів (залік)</td><td>Сумарна к-ть балів</td></tr>
<tr><td colspan="4">Змістовий модуль 1 (40 балів)</td><td colspan="4">Змістовий модуль 2 (40 балів)</td><td rowspan="3">20</td><td rowspan="3">100</td></tr>
<tr><td>Т1</td><td>Т2</td><td>Т3</td><td>Т4</td><td>Т5</td><td>Т6</td><td>Т7</td><td>Т8</td></tr>
<tr><td>10</td><td>10</td><td>10</td><td>10</td><td>10</td><td>10</td><td>10</td><td>10</td></tr>
</table>

Т1, Т2 ... Т8 – теми змістових модулів.

7 семестр

<table>
<tr><td colspan="8">Поточне оцінювання (аудиторна та самостійна робота)</td><td>Кількість балів (екзамен)</td><td>Сумарна к-ть балів</td></tr>
<tr><td colspan="4">Змістовий модуль 1 (40 балів)</td><td colspan="4">Змістовий модуль 2 (40 балів)</td><td rowspan="3">20</td><td rowspan="3">100</td></tr>
<tr><td>Т1</td><td>Т2</td><td>Т3</td><td>Т4</td><td>Т5</td><td>Т6</td><td>Т7</td><td>Т8</td></tr>
<tr><td>10</td><td>10</td><td>10</td><td>10</td><td>10</td><td>10</td><td>10</td><td>10</td></tr>
</table>

Т1, Т2 ... Т8 – теми змістових модулів.

Шкала оцінювання: національна та ECTS

Оцінка за національною шкалою	**Оцінка за шкалою ЄКТС**	
	Оцінка (бали)	**Пояснення за розширеною шкалою**
Відмінно	A (90-100)	відмінно
Добре	B (80-89)	дуже добре
	C (70-79)	добре
Задовільно	D (60-69)	задовільно
	E (50-59)	достатньо

Незадовільно	FX (35-49)	(незадовільно) з можливістю повторного складання
	F (1-34)	(незадовільно) з обов'язковим самостійним опрацюванням освітнього компоненту до перескладання

На оцінку "відмінно" заслуговує студент, який виявив всебічні, систематичні та глибокі знання, здатність самостійно виконувати завдання, передбачені програмою, ознайомлений з основною і додатковою літературою, рекомендованою програмою. Така оцінка передбачає також засвоєння студентом взаємозв'язку основних понять дисципліни та їх значення для набутої професії.

Оцінку "добре" ставлять студентові, який засвоїв навчально-програмовий матеріал у повному обсязі, успішно виконує передбачені програмою завдання, опрацював основну літературу, рекомендовану програмою, тобто студентові, який засвідчив систематичний характер знань із дисципліни і здатний до їх самостійного поповнення й оновлення у процесі подальшої навчальної роботи і професійної діяльності. На оцінку "задовільно" заслуговує студент, який виявив знання основного навчального матеріалу в обсязі, необхідному для подальшого навчання і майбутньої роботи за професією, здатний виконувати завдання, передбачені програмою, ознайомлений з основною літературою, рекомендованою програмою. Як правило, цю оцінку виставляють студентам, які припустилися огріхів у відповіді на іспиті та при виконанні екзаменаційних завдань, але продемонстрували спроможність усунути їх.

Оцінку "незадовільно" ставлять студентові, у знаннях якого є прогалини, який припустився принципових помилок у виконанні передбачених програмою завдань, тобто студентові, який неспроможний продовжити навчання чи приступити до професійної діяльності після закінчення вищого навчального закладу без додаткових занять з відповідної дисципліни.

Перелік питань для самоконтролю та підсумкового контролю навчальних досягнень студентів

6 семестр

1. Створення программи в .Net з та без використання методу Main. Приклади
2. Доступні типи в .Net. Приклади.
3. Умовні оператори та цикли. Приклади.
4. Масиви, колекції та робота з ними. Приклади.
5. Структури, класи, записи. Створення власних типів, приклади.

6. Оголошення та реалізація інтерфейсів, множинне наслідування. Явна, неявна реалізація та реалізація за замовчуванням.
7. Делегати, події, їх використання. Приклади.
8. Реалізація підтримки багатомовності та різних часових поясів. Приклади.
9. Робота з регулярними виразами. Приклади.
10. Типи серіалізаторів, їх використання приклади.
11. LINQ. Методи, взаємодія з джерелами, відкладене виконання. Приклади.
12. Основи роботи з TPL, Parallel, Task. Приклади.
13. Основи роботи з потокозахищеними колекціями. Приклади.
14. Типи додатків, для розробки в .Net.

7 семестр

1. ADO.NET: під'єднання, маніпуляція з об'єктами та даними, отримання даних, пул підключень, діагностика підключення. Приклади.
2. Entity Framework: провайдери, робота з реляційними та нереляційними базами даних. Підходи до зв'язування об'єктів та отримання інформації. Приклади.
3. ASP.NET MVC. Структура рішення, застосування, приклади реалізації.
4. ASP.NET Razor Pages. Структура рішення, застосування, приклади реалізації.
5. ASP. NET Web Api. Структура рішення, застосування, приклади реалізації.
6. Blazor. Структура рішення (сервер, клієнт-сервер, гібридне), застосування, приклади реалізації.
7. GraphQL. Структура рішення, застосування, приклади реалізації.
8. OData. Структура рішення, застосування, приклади реалізації.
9. gRPC. Структура рішення, застосування, приклади реалізації.
10. MAUI. Структура рішення, застосування, приклади реалізації.
11. Шифрування, хешування, підпис та верифікація засобами .Net. Приклади.
12. Typescipt. Переваги строгої типізації та раннього виявлення помилок, приклад використання.
13. Розгортання додатків та їх верифікація.

Зарахування результатів неформальної освіти

Зарахування результатів неформальної освіти проводиться у відповідності до "Порядку визнання у Чернівецькому національному університеті імені Юрія Федьковича результатів навчання, здобутих шляхом неформальної та/або інформальної освіти" (затверджено Вченою радою ЧНУ протокол № 16 від 25 листопада 2024 р.)

https://www.chnu.edu.ua/media/4g5fzssb/poriadok-vyznannia-rezultativ-navchannia-zdobutykh-shliakhom-neformalnoi-ta-abo-informalnoi-osvity.pdf.

Рекомендована література

Основна

1. Coding Clean, Reliable, and Safe REST APIs with ASP.NET Core 8: Develop Robust Minimal APIs with .NET 8 First Edition // Anthony Giretti, 2023 Apress.- 448pp.
2. ASP.NET 8 Best Practices: Explore techniques, patterns, and practices to develop effective large-scale .NET web apps // Jonathan Danylko, 2023 Packt Publishing.- 253pp.
3. Head First C#: A Learner's Guide to Real-World Programming with C# and .NET Core 4th Edition // Andrew Stellman, Jennifer Greene, 2021 O'Reilly Media.- 800 pp.
4. Pro C# 9 with .NET 5: Foundational Principles and Practices in Programming 10th ed. // Troelsen A., Japikse Ph., 2020 Apress.- 1411 pp.
5. C# 9.0 in a Nutshell: The Definitive Reference 1st Edition // Joseph Albahari, O'Reilly Media.- 1060 pp.
6. C# 9 and .NET 5 – Modern Cross-Platform Development: Build intelligent apps, websites, and services with Blazor, ASP.NET Core, and Entity Framework Core using Visual Studio Code, 5th Edition 5th ed. Edition // Mark J. Price, Packt.- 822 pp.

Допоміжна

1. ULTIMATE ASP.NET CORE WEB API // Marinko Spasojevic, Vladimir Pecanac, 2021 CodeMaze, 288 pp.
2. Розробка додатків засобами мови програмування C#: Навч.-метод. посібник для проведення лабораторних робіт для студентів вищих навчальних закладів спеціальності «системний аналіз» /Є.В.Івохін, М.Ф.Махно, О.Г.Піскунов. – К.: Видавничо-поліграфічний центр "Київський університет", 2021. – 100 с.
3. Miles Rob / C# Programming Edition 4.0 // Rob Miles Department of Computer Science University of Hull.- 200 pp.

Інформаційні ресурси

1. https://www.codecademy.com/learn/learn-c-sharp
2. https://visualstudio.microsoft.com/
3. https://learn.microsoft.com/
4. https://www.w3schools.com/cs/
5. https://dotnet.microsoft.com/

Політика академічної доброчесності

Дотримання політики щодо академічної доброчесності учасниками освітнього процесу при вивченні освітнього компонента регламентовано чинними документами ЧНУ:

- "Кодекс академічної доброчесності та етики Чернівецького національного університету імені Юрія Федьковича" https://www.chnu.edu.ua/universytet/normatyvni-dokumenty/kodeks-akademichnoi-dobrochesnosti-ta-etyky-chernivetskoho-natsionalnoho-universytetu-imeni-yuriia-fedkovycha/
- "Положення про виявлення та запобігання академічного плагіату у Чернівецькому національному університеті імені Юрія Федьковича" https://www.chnu.edu.ua/universytet/normatyvni-dokumenty/polozhennia-pro-vyiavlennia-ta-zapobihannia-akademichnomu-plahiatu-u-chernivetskomu-natsionalnomu-universyteti-imeni-yuriia-fedkovycha/
- "Політика використання штучного інтелекту в Чернівецькому національному університету імені Юрія Федьковича" https://www.chnu.edu.ua/universytet/normatyvni-dokumenty/polityka-vykorystannia-shtuchnoho-intelektu-v-chernivetskomu-natsionalnomu-universyteti-imeni-yuriia-fedkovycha-zi-zminamy-zatverdzhenymy-vchenoiu-radoiu-chnu-imeni-yuriia-fedkovycha/

Використання здобувачами освіти інструментів штучного інтелекту (ШІ) під час виконання навчальних завдань:

- *допускається* за умови дотримання принципів академічної доброчесності (здобувач освіти несе відповідальність за зміст, достовірність і результати поданої роботи, а у разі використання ШІ зобов'язаний зазначити факт і спосіб його застосування);
- *не допускається* використання ШІ як заміни самостійного виконання завдань або з метою порушення вимог академічної доброчесності.