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
Основи теорії алгоритмів
(назва навчальної дисципліни)

обов’язкова
(вказати: обов’язкова)

Освітньо-професійна програма Інформатика та інформаційні технології в освіті
(назва програми)

Спеціальність 014.09 Середня освіта (інформатика)
(вказати: код, назва)

Галузь знань 01 Освіта/Педагогіка
(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 «*Основи теорії алгоритмів*» складена відповідно до освітньо-професійної програми Інформатика та інформаційні технології в освіті

Розробник:
Івасюк Галина Петрівна, доцент кафедри математичного моделювання, кандидат фіз.-мат. наук, доцент

Викладач, що забезпечує читання даної навчальної дисципліни:
Івасюк Галина Петрівна, доцент кафедри математичного моделювання, кандидат фіз.-мат. наук, доцент

Погоджено з гарантом ОП _____ **Володимир ЛУЧКО**

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:

- розвиток логічного, аналітичного мислення;
- формування теоретичної бази знань студентів щодо структур даних, побудови алгоритмів з використанням структур даних та їх реалізації мовою програмування;
- розвиток уміння розв’язувати алгоритмічні задачі, користуючись відомими теоретичними положеннями, математичним апаратом, літературою та комп’ютерною технікою;
- доведення вивчення курсу до творчого рівня;
- бачення студентами можливостей використання набутих знань у їх майбутній професії.

Пререквізити. Для підвищення ефективності засвоєння курсу здобувач вищої освіти попередньо має вивчати курс «Програмування».

Результати навчання: Відповідно до освітньо-професійної програми підготовки бакалаврів галузі знань 01 – Освіта/Педагогіка за спеціальністю 014.09 – Середня освіта (інформатика) (освітньо-професійна програма «Інформатика та математика») вивчення дисципліни сприяє формуванню:

програмних компетентностей:

ПК3. Здатність до використання сучасних методів розробки та дослідження алгоритмів розв'язування задач у моделюванні об'єктів і процесів та реалізації цих алгоритмів сучасними мовами програмування;

ПК4. Здатність використовувати програмні засоби загального та спеціального призначення для розв’язання прикладних задач з інформатики;

ПК6. Здатність розв’язувати задачі шкільного курсу інформатики різного рівня складності, аналізувати та оцінювати ефективність розв'язку та формувати відповідні вміння в учнів;

програмних результатів навчання:

ПРН5. Визначає та застосовує методи розроблення та дослідження алгоритмів розв’язування задач з інформатики, описує і застосовує методи оцінювання ефективності алгоритмів;

ПРН9. Уміє реалізувати алгоритми розв’язання задач мовами програмування, вибирати й застосовувати інформаційно-комунікаційні технології; розв'язує задачі шкільного курсу інформатики різного рівня складності.

Опис навчальної дисципліни
Загальна інформація

Форма навчання	Рік підготовки	Семестр	Кількість		Кількість годин						Вид підсумкового контролю
			кредитів	всього годин	лекції	практичні	семінарські	лабораторні	самостійна робота	індивідуальні завдання	
Денна	3	5	3	90	14	–	–	30	46	–	екзамен
Заочна											

Структура змісту навчальної дисципліни

Назви змістових модулів і тем	Кількість годин											
	денна форма						*заочна форма*					
	усього	у тому числі					усього	у тому числі				
		л	п	лаб	інд	с.р.		л	п	лаб	інд	с.р.
1	2	3	4	5	6	7	8	9	10	11	12	13
Змістовий модуль 1. Базові структури даних та алгоритми обробки їх елементів												
Тема 1. Основні поняття. Базові структури даних.	13	2		4		7						
Тема 2. Найпростіші методи сортування	11	2		4		5						
Тема 3. Вказівники. Структуровані типи даних користувача	11	2		4		5						
Тема 4. Динамічна структура даних зв'язний список. Стек. Черга.	16	2		6		8						
Разом за ЗМ 1	51	8		18		25						
Змістовий модуль 2. Оптимізаційні алгоритми з використанням структур даних												
Тема 1. Бінарні дерева. Пошукові алгоритми на деревах.	13	2		4		7						
Тема2. Основні поняття теорії графів. Пошук в глибину та в ширину. Ейлерів та Гамільтонів графи.	12	2		4		6						
Тема 3. Пошук найкоротшого шляху між двома вершинами у зваженому графі.	14	2		4		8						
Разом за ЗМ2	39	6		12		21						
Усього годин	90	14		30		46						

Тематика лекційних занять з переліком питань

№	Назва теми з основними питаннями
1.	Основні поняття. Базові структури даних. ✓ Поняття алгоритму, його основні властивості та способи подання. ✓ Основні алгоритмічні конструкції. ✓ Поняття часової та просторової складності алгоритмів. ✓ Асимптотичні оцінки складності. ✓ Поняття структури даних, класифікація структур даних. ✓ Масиви та основні операції над ними.

2.	Найпростіші методи сортування ✓ Поняття сортування та його призначення. ✓ Основні характеристики алгоритмів сортування. ✓ Сортування вибором, обміном, вставками, підрахунком: принципи роботи та особливості реалізації. ✓ Аналіз часової складності найпростіших алгоритмів сортування.
3.	Вказівники. Структуровані типи даних користувача. Динамічна структура даних зв'язний список. Стек. Черга. ✓ Поняття вказівника та адресація даних. ✓ Структури як засіб створення користувацьких типів даних. ✓ Динамічне виділення та звільнення пам'яті. ✓ Поняття динамічних структур даних. ✓ Однозв'язні списки та основні операції над ними. ✓ Стек і черга: принципи організації, основні операції та особливості застосування.
4.	Бінарні дерева. Пошукові алгоритми на деревах. ✓ Поняття дерева та його основні характеристики. ✓ Бінарні дерева та їх властивості. ✓ Бінарні дерева пошуку: принципи побудови та організації. ✓ Операції пошуку, вставлення та видалення вузлів. ✓ Способи обходу бінарних дерев. ✓ Оцінювання складності основних операцій у бінарних деревах пошуку.
5.	Основні поняття теорії графів. Пошук в глибину та в ширину. ✓ Поняття графа, його основні елементи та характеристики. ✓ Орієнтовані й неорієнтовані, зважені й незважені графи. ✓ Способи подання графів: матриця суміжності та списки суміжності. ✓ Поняття зв'язності графа. ✓ Алгоритми пошуку в глибину та в ширину: принципи роботи, особливості реалізації та оцінювання складності.
6.	Ейлерів та Гамільтонів графи. ✓ Поняття маршруту, ланцюга та циклу в графі. ✓ Ейлерові ланцюги й цикли. ✓ Умови існування ейлерових циклів і ланцюгів. ✓ Гамільтонові шляхи та цикли. ✓ Поняття гамільтонового графа. ✓ Відмінності між ейлеровими та гамільтоновими графами. ✓ Задачі пошуку маршрутів на графах і складність їх розв'язування.
7.	Пошук найкоротшого шляху між двома вершинами у зваженому графі. ✓ Поняття шляху та його довжини у зваженому графі. ✓ Постановка задачі пошуку найкоротшого шляху. ✓ Алгоритм Дейкстри: принцип роботи, основні етапи та умови застосування. ✓ Визначення довжин найкоротших шляхів і відновлення маршруту. ✓ Оцінювання складності алгоритму.

Тематика лабораторних занять з переліком питань

№	Назва теми (завдання)
1.	**Основні поняття. Базові структури даних.** Розроблення та програмна реалізація алгоритмів опрацювання даних. Робота з масивами: пошук елементів, обчислення характеристик і перетворення даних. Тестування алгоритмів на різних наборах вхідних даних. Визначення та порівняння часової складності алгоритмів.
2.	**Найпростіші методи сортування** Програмна реалізація алгоритмів сортування вибором, обміном і вставками. Сортування масивів за зростанням і спаданням. Перевірка правильності роботи алгоритмів на різних наборах даних. Порівняння кількості операцій і часової ефективності методів сортування.
3.	**Динамічна структура даних зв'язний список. Стек. Черга.** Робота з вказівниками та динамічною пам'яттю. Створення й опрацювання однозв'язного списку: додавання, видалення та пошук елементів. Реалізація стека й черги. Виконання основних операцій над динамічними структурами даних. Перевірка коректності роботи структур і звільнення динамічно виділеної пам'яті.
4.	**Бінарні дерева. Пошукові алгоритми на деревах.** Створення та програмна реалізація бінарного дерева пошуку. Вставлення, пошук і видалення вузлів. Обхід дерева в прямому, симетричному та зворотному порядку. Виведення результатів обходу. Аналіз ефективності операцій пошуку залежно від структури дерева.
5.	**Основні поняття теорії графів. Пошук в глибину та в ширину.** Подання графів за допомогою матриці суміжності та списків суміжності. Програмна реалізація алгоритмів пошуку в глибину (DFS) та в ширину (BFS). Визначення порядку обходу вершин і пошук компонент зв'язності. Порівняння особливостей реалізації та складності алгоритмів обходу графів.
6.	**Ейлерів та Гамільтонів графи.** Визначення характеристик графів, необхідних для встановлення можливості існування ейлерових ланцюгів і циклів. Перевірка умов існування ейлерових циклів. Побудова ейлерових маршрутів. Дослідження гамільтонових шляхів і циклів на заданих графах. Розв'язування задач пошуку маршрутів та аналіз отриманих результатів.
7.	**Пошук найкоротшого шляху між двома вершинами у зваженому графі.** Подання зваженого графа у програмі. Програмна реалізація алгоритму Дейкстри для графів із невід'ємними вагами ребер. Визначення довжини найкоротшого шляху між заданими вершинами. Відновлення маршруту за результатами роботи алгоритму. Перевірка правильності результатів і аналіз складності алгоритму.

Завдання для самостійної роботи студентів

№	Назва теми	Завдання для самостійної роботи	К-сть год.
1.	Основні поняття. Базові структури даних.	Визначити часову складність алгоритмів пошуку максимального (мінімального, заданого) елемента для лінійного та бінарного пошуку. Визначити результат виконання заданого алгоритму для різних наборів вхідних даних. Підготувати приклади задач шкільного курсу інформатики, у яких доцільно використовувати масиви.	**7**
2.	Найпростіші методи сортування	Виконати сортування заданої послідовності методом обміну та вставками, відобразивши стан масиву після кожного проходу. Порівняти кількість порівнянь і перестановок для двох заданих послідовностей різного початкового впорядкування. Сформулювати навчальну задачу для учнів на пояснення одного з методів сортування.	**5**
3.	Вказівники. Структуровані типи даних користувача	Навести приклади оголошення структур для зберігання відомостей про студента та навчальну дисципліну мовою С.	**5**
4.	Динамічна структура даних зв’язний список. Стек. Черга.	Зобразити схему однозв'язного списку з п'яти елементів, позначивши інформаційні поля та вказівники. Визначити послідовність змін посилань під час додавання та видалення елемента зі списку. Виконати послідовність операцій над стеком і чергою (додавання, видалення, перегляд), відобразивши стан структур після кожної операції. Навести приклади використання стека та черги в навчальних і повсякденних ситуаціях.	**8**
5.	Бінарні дерева. Пошукові алгоритми на деревах.	Записати результати прямого та зворотного обходів побудованого дерева. Визначити послідовність порівнянь під час пошуку заданого елемента. Виконати вставку нового вузла та видалення вузла з одним нащадком. Порівняти кількість операцій пошуку в збалансованому та виродженому бінарних деревах.	**7**
6.	Основні поняття теорії графів. Пошук в глибину та в ширину. Ейлерів та Гамільтонів графи.	Побудувати граф за заданим набором вершин і ребер та скласти для нього матрицю суміжності. Подати цей граф у вигляді списків суміжності. Навести приклади використання графів для моделювання соціальних зв'язків, транспортних і комп'ютерних мереж. Встановити, чи містять графи Ейлерові цикли або ланцюги, обґрунтувавши відповідь. Визначити на заданих графах наявність Гамільтонового шляху або циклу. Сформулювати математичну модель задачі пошуку маршруту листоноші або обходу визначених об'єктів міста та пояснити, чи є вона Ейлеровою або Гамільтоновою задачею.	**6**

7.	Пошук найкоротшого шляху між двома вершинами у зваженому графі.	Побудувати зважений граф за заданою таблицею відстаней між вершинами. Виконати покрокове застосування алгоритму Дейкстри для знаходження найкоротшого шляху між двома заданими вершинами. Скласти практичну задачу на знаходження найкоротшого маршруту між населеними пунктами або об'єктами навчального закладу.	**8**

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

Методи навчання

Лекції, лабораторні роботи, тестування, аудиторне та дистанційне онлайн-навчання з використанням систем Moodle та Google Meet.

Методи навчання:

- вербальні методи (лекція, бесіда, пояснення, розповідь тощо);
- практичні методи (лабораторні роботи);
- наочні методи (демонстрація, ілюстрація);
- робота з інформаційними ресурсами: з навчально-методичною, нормативною літературою та інтернет-ресурсами;
- самостійна робота над індивідуальним завданням лабораторних робіт або за програмою навчальної дисципліни;
- дистанційне навчання з використанням відповідних онлайн-платформ.

Система контролю та оцінювання

Засобами оцінювання та демонстрування результатів навчання є усний контроль у вигляді індивідуального та фронтального опитування на лекціях та лабораторних заняттях, захист лабораторних робіт; письмовий контроль у вигляді підсумкового тестового опитування.

Формами поточного контролю є усна чи письмова (тестування, лабораторні роботи) відповідь студента.

Формою підсумкового контролю є екзамен.

Критерії оцінювання поточного та підсумкового контролю

Система оцінювання рівня навчальних досягнень ґрунтується на принципах ECTS та є накопичувальною. Протягом семестру студенти виконують 6 лабораторних робіт та проходять два тестування (по одному за кожен модуль). Лабораторні роботи оцінюються по 6 або 7 балів за кожну, тестування по 10 балів за кожне (див. таблицю нижче).

Виконуючи завдання лабораторної роботи, студент повинен оформити і завантажити для подальшої перевірки на сайт електронного навчання звіт разом із працездатними файлами програмної реалізації завдань ЛР (правила оформлення наведені на сторінці навчальної дисципліни на сайті).

50% балів, відведених на оцінювання ЛР, студент отримує за повністю правильно виконану ЛР та оформлений звіт. Решта 50% балів виставляється після захисту студентом виконаного звіту. 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 логічно неправильні твердження, то знімається до 50% балів від усієї суми балів за ЛР.

Максимальна кількість, яку можна набрати на підсумковому модулі (екзамені) – 40 балів. Екзамен проводиться у формі тестового опитування з використанням системи електронного навчання Moodle. Кожному студенту випадковим чином генерується 20 питань, правильна відповідь на які оцінюється в один бал за кожне. Сумарна максимальна кількість балів за виконання тестів становить 20 балів. Час виконання тесту – 30 хвилин.

Практичною частиною екзамену є розв'язання студентами двох задач, що містяться у базі даних даного курсу і також генерується випадковим чином. Виконання цього завдання оцінюється по 10 балів за кожне. Час виконання становить 30 хвилин.

Сумарна максимальна оцінка за екзамен становить 40 балів. Загальна оцінка за навчальну дисципліну визначається за результатами суми балів, набраних на змістових модулях під час семестру та підсумковому модулі згідно з нижченаведеною таблицею.

Розподіл балів, які отримують студенти

<table>
<tr><td colspan="8">Поточне тестування та самостійна робота</td><td>Підсумковий модуль (екзамен)</td><td>Сума</td></tr>
<tr><td colspan="4">Змістовий модуль 1</td><td colspan="4">Змістовий модуль 2</td><td rowspan="3">40</td><td rowspan="3">100</td></tr>
<tr><td>Т1-Т2</td><td>Т3</td><td>Т4</td><td>тести</td><td>Т1</td><td>Т2</td><td>Т3</td><td>тести</td></tr>
<tr><td>6</td><td>7</td><td>7</td><td>10</td><td>7</td><td>7</td><td>6</td><td>10</td></tr>
</table>

Шкала оцінювання: національна та ECTS

Оцінка за національною шкалою	**Оцінка за шкалою ECTS**	
	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

Перелік питань для самоконтролю та підсумкового контролю навчальних досягнень студентів

1. Що таке алгоритм? Які його основні властивості?
2. Які існують способи подання алгоритмів?
3. Які основні алгоритмічні конструкції використовуються під час розроблення алгоритмів?
4. Що характеризують асимптотичні оцінки складності алгоритмів?
5. Які основні класи асимптотичної складності алгоритмів?
6. Що таке структура даних? Які існують основні класифікації структур даних?
7. Які основні операції виконуються над масивами?

8. Що таке сортування даних і які його основні характеристики?
9. У чому полягають принципи сортування вибором, обміном і вставками?
10. У чому полягає принцип сортування підрахунком?
11. Які переваги та недоліки найпростіших методів сортування?
12. Що таке вказівник і яке його призначення в мові програмування С?
13. Які операції можна виконувати над вказівниками?
14. Що таке структура в мові С та як оголошуються користувацькі структуровані типи даних?
15. Яке призначення ключового слова typedef?
16. Що таке динамічне виділення та звільнення пам'яті? Які функції мови С для цього використовуються?
17. Що таке динамічні структури даних і чим вони відрізняються від статичних?
18. Що таке однозв'язний список? Які основні операції виконуються над ним?
19. Що таке стек? Який принцип покладено в основу його функціонування?
20. Що таке черга? Які основні операції виконуються над нею?
21. У чому полягають відмінності між стеком і чергою та які задачі вони дають змогу розв'язувати?
22. Що таке дерево та які його основні характеристики?
23. Що таке бінарне дерево та які його властивості?
24. Що таке бінарне дерево пошуку та які принципи його побудови?
25. Як виконуються операції пошуку, вставлення та видалення вузлів у бінарному дереві пошуку?
26. Які існують способи обходу бінарних дерев? У чому полягають їх відмінності?
27. Як залежить ефективність пошуку від структури бінарного дерева?
28. Що таке граф? Які його основні елементи та характеристики?
29. Чим відрізняються орієнтовані та неорієнтовані, зважені та незважені графи?
30. Які існують способи подання графів у пам'яті комп'ютера?
31. У чому полягають особливості матриці суміжності та списків суміжності?
32. Що таке зв'язний граф і компонента зв'язності?
33. У чому полягає принцип роботи алгоритму пошуку в глибину (DFS)?
34. У чому полягає принцип роботи алгоритму пошуку в ширину (BFS)?
35. Які структури даних використовуються під час реалізації алгоритмів DFS і BFS?
36. Яка часова складність алгоритмів пошуку в глибину та в ширину?
37. Що таке маршрут, ланцюг і цикл у графі?
38. Що таке ейлерів ланцюг та ейлерів цикл? Які умови їх існування?
39. Що таке гамільтонів шлях і гамільтонів цикл?
40. У чому полягають відмінності між ейлеровими та гамільтоновими графами?
41. Які практичні задачі можна розв'язувати за допомогою ейлерових і гамільтонових графів?
42. Що таке шлях у зваженому графі та як визначається його довжина?
43. У чому полягає задача пошуку найкоротшого шляху між двома вершинами графа?
44. У чому полягає принцип роботи алгоритму Дейкстри?
45. Які основні етапи виконання алгоритму Дейкстри?
46. За яких умов можна застосовувати алгоритм Дейкстри?
47. Як визначаються найкоротші відстані та відновлюється маршрут за результатами роботи алгоритму Дейкстри?
48. Яка часова складність алгоритму Дейкстри, його переваги та обмеження?

Зарахування результатів неформальної/інформальної освіти

Здобувачі вищої освіти має право на участь у неформальній/інформальній освіті.

У межах поточного контролю можуть визнаватися результати неформальної/інформальної освіти за умови наявності сертифікату або освітньої декларації про результати неформальної/інформальної освіти з питань, що відповідає тематиці курсу («Порядок визнання у Чернівецькому національному університеті імені Юрія Федьковича результатів навчання, здобутих шляхом неформальної та/або інформальної освіти», https://www.chnu.edu.ua/media/4g5fzssb/poriadok-vyznannia-rezultativ-navchannia-zdobutykh-shliakhom-neformalnoi-ta-abo-informalnoi-osvity.pdf).

Студентам можуть бути зараховані додаткові бали (максимум 5 балів), отримані через неформальну освіту, до загальної суми балів, набраної з освітньої компоненти, за умови, що результати з проблеми, за якою відбувалося навчання, відповідають тематиці курсу.

Рекомендована література

1. Томас Х. Кормен, Чарльз И. Лейзерсон, Рональд Л. Рівест, Кліффорд Штайн Алгоритми: побудова й аналіз, 3-е видання, том 1. Вильямс, Україна, 2020. 648 с.
2. Томас Х. Кормен, Чарльз И. Лейзерсон, Рональд Л. Рівест, Кліффорд Штайн Алгоритми: побудова й аналіз, 3-е видання, том 2. Диалектика, Україна, 2020. 664 с.
3. Т.П.Караванова. Теорія алгоритмів. Частина 1. Необчислювальні алгоритми. Навч.посібник. Чернівці: Чернівецький нац.ун-т, 2022. 268 с.
4. Т.П.Караванова. Теорія алгоритмів. Частина 2. Обчислювальні алгоритми. Навч.посібник. Чернівці: Чернівецький нац.ун-т, 2022. 288 с.
5. Alfred V. Aho, John E. Hopcroft, Jeffrey D.Ullman. Data Structures and Algorithms. Publisher: Pearson, 1st edition (January 1, 1983). 448 pages.
6. Niklaus Wirth Algorithms + Data Structures = Programs. Publisher: Prentice Hall; 1st edition (February 1, 1976). 366 pages.
7. Караванова Т.П. Методика розв'язування алгоритмічних задач. Побудова алгоритмів: Навчально-методичний посібник для вчителів .Кам'янець-Подільський: Аксіома, 2014, 344 с.
8. Караванова Т.П. Інформатика: методи побудови алгоритмів та їх аналіз: необчислювальні алгоритми: Навч. посіб. для 9-10 кл. із поглибл. вивч. інф-ки К.: Генеза. 2007. 216 с.
9. Програмування. Частина 2. Навчальний посібник / Укл.: Івасюк Г.П., Фратавчан Т.М., Івасюк Р.В. Чернівці: Чернівецький нац. ун-т, 2025. 164 с.
10. Основи теорії алгоритмів: лабораторний практикум. Укл.: Г.П. Івасюк, Т.М.Фратавчан, Р.В.Івасюк, Чернівці: Чернівецький нац.ун-т, 2026. 110 с.

Інформаційні ресурси

1. https://moodle.chnu.edu.ua/course/view.php?id=4706 – розміщення курсу на платформі https://moodle.chnu.edu.ua/.
2. https://apps.prometheus.org.ua/learning/course/course-v1:KPI+Algorithms101+2015_Spring/home

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 університеті 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;
- *додаткові обмеження чи вимоги* щодо використання ШІ *можуть визначатися* викладачем в освітньому процесі для конкретних видів навчальної діяльності відповідно до особливостей дисципліни.