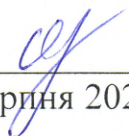


МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
ЕКОНОМІКО-ТЕХНОЛОГІЧНИЙ ІНСТИТУТ ІМЕНІ Р.ЕЛЬВОРТІ

Кафедра інформаційних технологій

"ЗАТВЕРДЖУЮ"  
Завідувач кафедри

 /О.П. Бондар/  
30 серпня 2024 року

РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

**«ТЕОРІЯ АЛГОРИТМІВ»**

Спеціальність: 122 Комп'ютерні науки

Освітньо-професійна програма: Комп'ютерні науки

Нормовані дані Форма навчання	Курс	Семестр	Всього годин за планом	Кількість кредитів ECTS	Всього аудит (год.)	Аудиторних годин, (у тому числі КЗ)			Самостійна робота (год.)	Курсове проектування (семестр/ кредити)	Контрольний підсумок (семестр)	
						Лекції	Лабораторні роботи	Практичні заняття			Екзамен	Залік
Денна	1	1	105	3,5	68	34		34	37		1	

Робочу програму складено на основі освітньо-професійної програми за спеціальністю: 122 Комп'ютерні науки

Робочу програму складено: доцент Паращук С.Д.

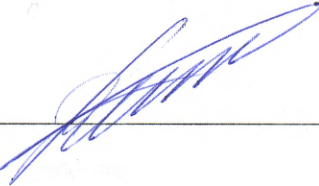
Робочу програму затверджено на засіданні кафедри інформаційних технологій.

Протокол № 1 від "30" серпня 2024 року

Завідувач кафедри  /О.П. Бондар/

Схвалено Вченою радою ЕТІ ім. Р.Ельворті

Протокол № 15 від "24" вересня 2024 року

Голова Вченої ради  /Штець Т.Ф./

## 1. ОПИС НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Найменування показників	Галузь знань, спеціальність, ступінь вищої освіти	Характеристика навчальної дисципліни	
		Денна форма навчання	Заочна форма навчання
Кількість кредитів – 3,5	Галузь: 12– Комп’ютерні науки Спеціальність або освітня програма 122 Комп’ютерні науки	Статус дисципліни нормативна	
Змістових модулів - 5		Рік підготовки	
Індивідуальне завдання студента		1	
Загальна кількість годин -105		Семестр	
Тижневих годин для денної форми навчання: аудиторних: 1 семестр – 4 години	Ступінь вищої освіти: бакалавр	1	
		Лекції (год.)	
		34	
		Практичні, семінарські (год.)	
		34	
		Лабораторні (год.)	
		-	-
		Самостійна робота (год.)	
		37	
		Індивідуальне завдання (год.)	
-	-		
Вид контролю:			
1 сем – екзамен			

### Примітка.

Співвідношення кількості годин аудиторних занять та самостійної роботи становить (%): 60% до 40% .

## 2. Мета та завдання навчальної дисципліни

### 2.1. МЕТА ВИКЛАДАННЯ ДИСЦИПЛІНИ

Структури даних і алгоритми є фундаментом сучасного програмування. Оволодіння технологією побудови і аналізу алгоритмів є важливою передумовою створення ефективного програмного забезпечення.

Метою викладання дисципліни є формування знань, вмінь і навичок у студентів, необхідних для ефективного алгоритмічного розв’язування практичних задач. Студенти повинні навчитись створювати і аналізувати алгоритми, ознайомитись із найчастіше використовуваними класичними алгоритмами та їх ефективністю.

### 2.2. ЗАВДАННЯ ВИВЧЕННЯ ДИСЦИПЛІНИ

Вивчення теоретичних понять теорії алгоритмів та ознайомлення з основними відомими алгоритмами, опанування базових методів розробки алгоритмів та аналізу ефективності алгоритмів.

2.3. За результатами вивчення дисципліни здобувач повинен опанувати наступні компетентності:

**Інтегральна компетентність.** Здатність розв’язувати складні спеціалізовані задачі та практичні проблеми у галузі комп’ютерних наук або у процесі навчання, що передбачає застосування теорій та методів інформаційних технологій і характеризується комплексністю та

невизначеністю умов.

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.

ЗК4. Здатність спілкуватися державною мовою як усно, так і письмово.

ЗК6. Здатність вчитися й оволодівати сучасними знаннями.

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

ЗК8. Здатність генерувати нові ідеї (креативність).

СК3. Здатність до логічного мислення, побудови логічних висновків, використання формальних мов і моделей алгоритмічних обчислень, проектування, розроблення й аналізу алгоритмів, оцінювання їх ефективності та складності, розв'язності та нерозв'язності алгоритмічних проблем для адекватного моделювання предметних областей і створення програмних та інформаційних систем.

2.4. За результатами вивчення навчальної дисципліни студент повинен:

**знати:**

- поняття алгоритму та його властивості, базові алгоритмічні конструкції;
- формальні моделі алгоритмів;
- методи обчислення ефективності алгоритму;
- структури даних, які часто використовуються при побудові алгоритмів;
- основні алгоритми на графах;
- різні алгоритми сортування та їх ефективність;
- методи розробки алгоритмів;

**вміти:**

- застосовувати теоретичні знання в практиці побудови алгоритмів;
- здійснити аналіз алгоритму та оцінити його ефективність.

2.5. Програмні результати навчання

ПР1. Застосовувати знання основних форм і законів абстрактно-логічного мислення, основ методології наукового пізнання, форм і методів вилучення, аналізу, обробки та синтезу інформації в предметній області комп'ютерних наук.

ПР5. Проектувати, розробляти та аналізувати алгоритми розв'язання обчислювальних та логічних задач, оцінювати ефективність та складність алгоритмів на основі застосування формальних моделей алгоритмів та обчислюваних функцій.

ПР9. 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.

## 2.6. Структурно-логічне місце дисципліни в освітній програмі

Попередні дисципліни:	Наступні дисципліни:
Шкільні курси математики та інформатики	Програмування
	Об'єктно-орієнтоване програмування

Програма навчальної дисципліни складається з п'яти модулів

## 3. Програма навчальної дисципліни

**Змістовий модуль 1. Основи алгоритмізації.**

**Тема 1.1.** Поняття алгоритму. Подання алгоритмів.

1. Поняття алгоритму.

2. Властивості алгоритмів.
3. Способи подання алгоритмів.
4. Блок-схеми та їхні елементи.

**Тема 1.2.** Типи обчислювальних процесів.

1. Структурна теорема. Технологія структурного програмування.
2. Базові алгоритмічні структури: слідування, розгалуження, повторення.
3. Лінійні алгоритми.
4. Реалізація алгоритму з розгалуженнями.
5. Реалізація і використання алгоритмів з циклами.

**Тема 1.3.** Оцінки складності алгоритмів.

1. Поняття оцінки складності.
2. Види оцінок.
3. Класифікація алгоритмів за складністю.

**Змістовий модуль 2. Формальні моделі алгоритмів.**

**Тема 2.1.** Алгоритмічні моделі.

1. Машини з натуральнозначними регістрами.
2. МНР-програми.
3. Машини Тьюрінга.
4. Нормальні алгоритми А.А. Маркова.

**Тема 2.2.** Алгоритми та обчислювальні функції.

1. Загальні визначення.
2. Примітивно рекурсивні функції.
3. Частково рекурсивні функції.
4. Теза Черча.

**Тема 2.3.** Нумерації.

1. Нумерація множин натуральних чисел.
2. Нумерація множин обчислювальних функцій.
3. Канторівська нумерація пар натуральних чисел.

**Змістовий модуль 3. Сортування і пошук.**

**Тема 3.1.** Сортування.

1. Задача сортування. Методи сортування.
2. Прості алгоритми сортування.
3. Швидке сортування.
4. Пірамідальне сортування.
5. Вибір методу сортування.

**Тема 3.2.** Пошук.

1. Задача пошуку. Методи пошуку.
2. Послідовний пошук.
3. Бінарний пошук.

**Тема 3.3.** Пошук з поверненням.

1. Реалізація пошуку з поверненням.
2. Алфа – бета відсікання.
3. Метод гілок та меж.

**Тема 3.4.** Хешування.

1. Пряма адресація.
2. Хеш-таблиці.
3. Оцінка ефективності хеш-функцій.
4. Відкрита адресація.

**Змістовий модуль 4. Алгоритми на графах.**

**Тема 4.1.** Деревя.

1. Поняття дерева. Двійкові дерева.
2. Представлення дерев.
3. Алгоритми обходу дерев.
4. Двійкові дерева.
5. Дерева виразів.
6. Бінарні дерева пошуку.
7. Коди Хафмана.

**Тема 4.2.** Основні алгоритми на графах.

1. Представлення графів.
2. Пошук в ширину.
3. Пошук в глибину.
4. Топологічне сортування.

**Тема 4.3.** Стовбурні дерева.

1. Побудова стовбурного дерева.
2. Алгоритми Крускала і Пріма.

**Тема 4.4.** Найкоротші шляхи.

1. Алгоритм Дейкстри.
2. Алгоритм Беллмана-Форда.
1. Алгоритм Флойда-Уоршала.

**Тема 4.5.** Максимальний потік.

1. Потоки в мережах.
2. Метод Форда-Фалкерсона.
3. Максимальне паросполучення у дводольному графі.

**Змістовий модуль 5. Методи аналізу та побудови алгоритмів.**

**Тема 5.1.** Методи аналізу алгоритмів.

1. Ефективність алгоритмів.
2. Аналіз рекурсивних програм.
3. Розв'язування рекурентних співвідношень.
4. Загальний розв'язок великого класу рекурентних співвідношень.

**Тема 5.2.** Методи розробки алгоритмів.

1. Метод «розділай і пануй».
2. Динамічне програмування.
3. Жадібний метод. Матроїди.

**Тема 5.3.** NP-повнота алгоритмів.

1. Поліноміальний час.
2. Перевірка належності мови і клас NP.
3. NP-повнота і звідність.
4. Доведення NP-повноти.
5. NP-повні задачі.

**4. Структура навчальної дисципліни**

Назви змістових модулів і тем	Кількість годин					
	усього	у тому числі				
		л	п	лаб	інд	с.р.
1	2	3	4	5	6	7
<b>Змістовий модуль 1. Основи алгоритмізації.</b>						
<b>Тема 1.1.</b> Поняття алгоритму. Подання алгоритмів.	<b>3</b>	<b>2</b>				<b>1</b>
<b>Тема 1.2.</b> Типи обчислювальних	<b>11</b>	<b>2</b>	<b>8</b>			<b>1</b>

процесів.						
<b>Тема 1.3.</b> Оцінки складності алгоритмів.	<b>3</b>	<b>2</b>				<b>1</b>
<b>Разом за змістовим модулем 1</b>	<b>17</b>	<b>6</b>	<b>8</b>			<b>3</b>
<b>Змістовий модуль 2. Формальні моделі алгоритмів.</b>						
<b>Тема 2.1.</b> Алгоритмічні моделі.	<b>12</b>	<b>4</b>	<b>6</b>			<b>2</b>
<b>Тема 2.2.</b> Алгоритми та обчислювальні функції.	<b>6</b>	<b>2</b>	<b>2</b>			<b>2</b>
<b>Тема 2.3.</b> Нумерації.	<b>4</b>	<b>2</b>				<b>2</b>
<b>Разом за змістовим модулем 2</b>	<b>22</b>	<b>8</b>	<b>8</b>			<b>6</b>
<b>Змістовий модуль 3. Сортування і пошук.</b>						
<b>Тема 3.1.</b> Сортування.	<b>7</b>	<b>2</b>	<b>4</b>			<b>1</b>
<b>Тема 3.2.</b> Пошук.	<b>5</b>	<b>1</b>	<b>2</b>			<b>2</b>
<b>Тема 3.3.</b> Пошук з поверненням.	<b>3</b>	<b>1</b>				<b>2</b>
<b>Тема 3.4.</b> Хешування.	<b>7</b>	<b>2</b>	<b>2</b>			<b>3</b>
<b>Разом за змістовим модулем 3</b>	<b>22</b>	<b>6</b>	<b>8</b>			<b>8</b>
<b>Змістовий модуль 4. Алгоритми на графах.</b>						
<b>Тема 4.1.</b> Древа.	<b>5</b>	<b>2</b>	<b>2</b>			<b>1</b>
<b>Тема 4.2.</b> Основні алгоритми на графах.	<b>5</b>	<b>2</b>	<b>2</b>			<b>1</b>
<b>Тема 4.3.</b> Стовбурні дерева.	<b>5</b>	<b>2</b>	<b>2</b>			<b>1</b>
<b>Тема 4.4.</b> Найкоротші шляхи.	<b>5</b>	<b>2</b>	<b>2</b>			<b>1</b>
<b>Тема 4.5.</b> Максимальний потік.	<b>7</b>	<b>1</b>				<b>6</b>
<b>Разом за змістовим модулем 4</b>	<b>27</b>	<b>9</b>	<b>8</b>			<b>10</b>
<b>Змістовий модуль 5. Методи аналізу та побудови алгоритмів.</b>						
<b>Тема 5.1.</b> Методи аналізу алгоритмів.	<b>6</b>	<b>2</b>	<b>2</b>			<b>2</b>
<b>Тема 5.2.</b> Методи розробки алгоритмів.	<b>4</b>	<b>2</b>				<b>2</b>
<b>Тема 5.3.</b> NP-повнота алгоритмів.	<b>7</b>	<b>1</b>				<b>6</b>
<b>Разом за змістовим модулем 5</b>	<b>17</b>	<b>5</b>	<b>2</b>			<b>10</b>
<b>Разом</b>	<b>105</b>	<b>34</b>	<b>34</b>			<b>37</b>

#### 5. Теми семінарських занять

№ з/п	Назва теми	Кількість годин
1	Не передбачено	

#### 6. Теми практичних занять

№ з/п	Назва теми	Кількість годин
<b>Змістовий модуль 1. Основи алгоритмізації.</b>		<b>8</b>
1.	Типи обчислювальних процесів. 1. Лінійні алгоритми. 2. Розгалуження. 3. Цикли.	8
<b>Змістовий модуль 2. Формальні моделі алгоритмів.</b>		<b>8</b>
2.	Алгоритмічні моделі. 1. МНР-програми.	6

	2. Машини Тюрінга. 3. Нормальні алгоритми Маркова.	
3.	Алгоритми та обчислювальні функції. Рекурсивні функції	2
<b>Змістовий модуль 3. Сортування і пошук.</b>		<b>8</b>
4.	Сортування. 1. Прості алгоритми сортування. 2. Швидке та пірамідальне сортування.	4
5.	Пошук. Алгоритми послідовного та бінарного пошуку	2
6	Хешування	2
<b>Змістовий модуль 4. Алгоритми на графах.</b>		<b>8</b>
7.	Дерева. Алгоритми обходу дерев. Дерева виразів	2
8.	Основні алгоритми на графах. Представлення графів. Пошук в ширину. Пошук в глибину. Топологічне сортування.	2
9	Стовбурні дерева. Побудова стовбурного дерева. Алгоритми Крускала і Пріма.	2
10.	Найкоротші шляхи. Алгоритм Дейкстри.	2
<b>Змістовий модуль 5. Методи аналізу та побудови алгоритмів.</b>		<b>2</b>
11.	Методи аналізу алгоритмів. Ефективність алгоритмів. Аналіз рекурсивних програм. Розв'язування рекурентних співвідношень.	2
	<b>Разом</b>	<b>34</b>

#### 7. Теми лабораторних занять

№ з/п	Назва теми	Кількість годин
1	Не передбачено	

#### 8. Самостійна робота

№ з/п	Назва теми	Кількість годин
1	<b>Тема 4.5.</b> Максимальний потік. 1. Потоки в мережах. 2. Метод Форда-Фалкерсона. 3. Максимальне паросполучення у дводольному графі.	6
2	<b>Тема 5.3.</b> NP-повнота алгоритмів. 1. Поліноміальний час. 2. Перевірка належності мови і клас NP. 3. NP-повнота і звідність. 4. Доведення NP-повноти. 5. NP-повні задачі.	6
3	<b>Опрацювання лекцій та підготовка до практичних занять</b>	25
	<b>Разом</b>	<b>37</b>

#### 9. Індивідуальні завдання

Не передбачено

#### 10. Методи навчання

Лекції із застосування мультимедійних технологій навчання, практичні заняття для формування навичок розв'язування задач, консультації.

#### 11. Методи контролю



Оцінювання якості знань студентів здійснюється шляхом поточного, підсумкового (семестрового) контролю за 100-бальною шкалою оцінювання, за шкалою ECTS та національною шкалою оцінювання.

**Поточний контроль** - оцінювання засвоєння студентом навчального матеріалу під час проведення практичних занять, виконання індивідуальних домашніх завдань, консультацій. Результати поточного контролю заносяться в журнал у балах (5, 4, 3, 2, 1). Сума балів за поточний контроль максимально дорівнює 60.

**Підсумковий (семестровий) контроль** – комплексне оцінювання якості засвоєння навчального матеріалу дисципліни на екзамені. Сума балів за екзамен максимально дорівнює 40.

## 12. Розподіл балів, які отримують студенти

Поточне оцінювання та самостійна робота					Екзамен	Сума
Змістовий модуль №1	Змістовий модуль №2	Змістовий модуль №3	Змістовий модуль №4	Змістовий модуль №5		
15	15	10	15	5	40	100

## Шкала оцінювання: національна та ECTS

Сума балів за всі види навчальної діяльності	Оцінка ECTS	Оцінка за національною шкалою	
		для екзамену, курсового проекту (роботи), практики	для заліку
90 – 100	<b>A</b>	відмінно	зараховано
82-89	<b>B</b>	добре	
74-81	<b>C</b>		
64-73	<b>D</b>	задовільно	
60-63	<b>E</b>		
35-59	<b>FX</b>	незадовільно з можливістю повторного складання	не зараховано з можливістю повторного складання
0-34	<b>F</b>	незадовільно з обов'язковим повторним вивченням дисципліни	не зараховано з обов'язковим повторним вивченням дисципліни

## 13. Методичне забезпечення

1. Навчально-методичний комплекс.
2. Тексти лекцій (в електронному варіанті).

## 14. Рекомендована література

### Основна

1. Матвієнко М.П. Теорія алгоритмів. Навчальний посібник. К.: Вид-во Ліра, 2018. 340 с.
2. Бородкіна І.Л., Бородкін Г.О. Теорія алгоритмів. Навчальний посібник. К.: Вид-во «Центр навчальної літератури», 2019. 184 с.
3. Прийма С.М. Теорія алгоритмів: Навчальний посібник. Мелітополь: ФОП Однорог Т.В., 2018. 116 с.
4. Кублій Л.І. Алгоритми та структури даних. Основи алгоритмізації. Підручник. К.: КПІ ім. Ігоря Сікорського, Електронне мережне навчальне видання, 2022. 528 с.
5. Крєневич А.П. Алгоритми і структури даних. Підручник. – К.: ВПЦ "Київський Університет", 2021. – 200 с.

6. Паращук С.Д., Сурков К.Ю., Ізвалов О.В. Практикум із програмування мовою С++: Навчальний посібник. Кропивницький: ЕТІ імені Роберта Ельворті, 2023, 260 с.

#### Допоміжна

1. Глибовець М.М. Основи комп'ютерних алгоритмів. – К.: Вид. дім “КМ Академія”, 2003. – 452 с.

#### 15. Інформаційні ресурси

1. Кевін Славін: Як алгоритми формують наш світ. URL: <https://lnnk.in/dkcf>
2. Quick-sort with Hungarian folk dance. URL: <https://www.youtube.com/watch?v=ywWBy6J5gz8>
3. Веселкова візуалізація алгоритмів сортування. URL: <https://echo.lviv.ua/dev/6534>.
4. 8 сервісів для візуалізації алгоритмів. URL: <https://echo.lviv.ua/dev/6647#more-6647>.
5. Сортування бульбашкою. URL: <https://lnnk.in/gubo>. 6. Сортування змішуванням. URL: <https://lnnk.in/jMa7>