

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Львівський національний університет імені Івана Франка
Факультет прикладної математики та інформатики
Кафедра програмування

Затверджено

На засіданні кафедри програмування
факультету прикладної математики
Львівського національного університету
імені Івана Франка
(протокол № 1 від 29 серпня 2025р.)



Зав. кафедри к. ф.-м. н., доц. Ярошко С. А.

Силабус навчальної дисципліни
«Алгоритми і структури даних»
що викладається в межах
ОПП «Середня освіта (Інформатика)»
для здобувачів першого (бакалаврського) рівня вищої освіти
з предметної спеціальності **А4.09 Середня освіта (Інформатика)**
галузі знань **А Освіта/Педагогіка**

Львів 2025 р.

Назва дисципліни	Алгоритми і структури даних
Адреса викладання дисципліни	Львівський національний університет імені Івана Франка, вул. Університетська 1, м. Львів, Україна, 79000
Факультет та кафедра, за якою закріплена дисципліна	Факультет прикладної математики та інформатики, кафедра програмування
Галузь знань, шифр та назва спеціальності	Галузь знань: А Освіта/Педагогіка Предметна спеціальність: А4.09 Середня освіта (Інформатика)
Викладачі дисципліни	Літинський Святослав Володимирович, к. ф.-м. н., доцент, доцент кафедри програмування
Контактна інформація викладачів	Електронна пошта: svyatoslav.litynskyi@lnu.edu.ua веб-сторінка: https://ami.lnu.edu.ua/employee/litynskyi
Консультації з питань навчання по дисципліні відбуваються	Консультації проводять раз на тиждень згідно з оприлюдненим розкладом консультацій викладача. Можливі он-лайн консультації через Microsoft Teams. Для погодження часу он-лайн консультацій слід писати на електронну пошту викладача.
Сторінка курсу	https://ami.lnu.edu.ua/course/algorithms-and-data-structures-informatics
Інформація про дисципліну	Курс “Алгоритми і структури даних” є нормативною дисципліною для освітньо-професійної програми «Середня освіта (Інформатика)» першого (бакалаврського) рівня вищої освіти з предметної спеціальності А4.09 Середня освіта (Інформатика), яка викладається в 2-му семестрі в обсязі 4 кредитів (за Європейською Кредитно-Трансферною Системою ECTS).
Коротка анотація дисципліни	У курсі розглядається основні структури даних на основі багаторівневого відображення. Наводиться класифікація абстрактних структур даних та структур зберігання даних. Вивчаються абстрактні структури: масив, таблиця, рядок та динамічні структури (стек, черга, дек), списки, дерева та графи. Розглядаються відображення цих структур у структури зберігання даних і базові операції над ними.
Мета та цілі дисципліни	Метою нормативної дисципліни «Алгоритми і структури даних» є навчити студента: <ul style="list-style-type: none"> • базовим структурам даних та їх перевагам та недолікам; • виконувати часову та ємнісну оцінку складності алгоритмів з різними структурами даних; • застосовувати основні структури даних у прикладних задачах.
Література для вивчення дисципліни	<i>Основна література</i> <ol style="list-style-type: none"> 1. Костів О. Структури даних. Видавничий центр ЛНУ імені Івана Франка . 2005. 2. Томас Кормен, Чарльз Лейзерсон, Рональд Рівест, Кліффорд Стайн. Вступ до алгоритмів. К: – К.І.С., 2019. – 1288 С. 3. Крєневич А.П. Алгоритми і структури даних. Підручник. – К.: ВПЦ "Київський Університет", 2021. – 200 с. https://www.mechmat.univ.kiev.ua/wp-content/uploads/2021/09/pidruchnyk-alhorytmy-i-strukturny-danykh.pdf <i>Додаткова література</i> <ol style="list-style-type: none"> 4. Ярошко С.А. Методи розробки алгоритмів. Програмування мовою С++: Навчальний посібник / С.А. Ярошко, О.С. Ярошко – Львів: ЛНУ імені Івана Франка, 2022. – 248 с. [електронна версія: https://lnuitutor.github.io/] 5. Бугаєва Л. М. Алгоритми та структури даних. Комп’ютерний практикум Навчальний посібник / Л. М. Бугаєва, Д. О. Ковалюк – Київ: КПІ ім. Ігоря Сікорського, 2022. – 34с. https://ela.kpi.ua/server/api/core/bitstreams/e07c0ac4-a6a0-4412-b2b1-3b21bcebaa2e/content
Обсяг курсу	4 кредити ЄКТС – 120 годин. З них 32 годин лекцій, 32 години лабораторних занять та 56 годин самостійної роботи

<p>Очікувані результати навчання</p>	<p>Після завершення цього курсу студент буде:</p> <p><i>знати</i></p> <ul style="list-style-type: none"> про існування, способи реалізації та характеристики часової та ємнісної складності різних структур даних. <p><i>вміти</i></p> <ul style="list-style-type: none"> Написати самостійно та використати існуючу бібліотеку з різними структурами даних.
<p>Компетентності</p>	<p>Інтегральна: ІК - Здатність розв'язувати складні спеціалізовані практичні завдання в галузі середньої освіти, що передбачає застосування концептуальних методів освітніх наук, знань з інформатики, педагогіки, психології, теорії та методики навчання інформатики і характеризується комплексністю та невизначеністю умов організації освітнього процесу в закладах загальної середньої освіти</p>
<p>Програмні результати навчання</p>	<p>Загальні: ЗК4 - Здатність орієнтуватися в інформаційному просторі, здійснювати пошук, аналіз та обробку інформації з різних джерел, ефективно використовувати цифрові ресурси та технології в освітньому процесі. Фахові компетентності спеціальності: ПК 4 - Здатність використовувати програмні засоби загального та спеціального призначення для розв'язання прикладних задач з інформатики, ПК9 - Здатність до створення концептуальної, логічної та фізичної моделей проектування систем керування базами даних..</p>
<p>Програмні результати навчання</p>	<p>ПРН 7 Демонструє знання основ фундаментальних і прикладних наук інформатики та програмування, оперує базовими категоріями та поняттями предметної області спеціальності, ПРН 17 Визначає та застосовує методи розроблення та дослідження алгоритмів розв'язування задач з інформатики, описує і застосовує методи оцінювання ефективності алгоритмів</p>
<p>Ключові слова</p>	<p>Алгоритм, тип даних, структура даних, часова складність, просторова складність, стек, дек, дерево, граф, хеш таблиця, список.</p>
<p>Формат курсу</p>	<p>Очний: проведення лекцій, лабораторних робіт та консультацій в приміщеннях університету, а в умовах карантину – онлайнний на платформі Microsoft Teams</p>
<p>Теми</p>	<p>Лекції Вступ Багаторівневе відображення даних</p> <ul style="list-style-type: none"> Змістовний рівень опису даних Абстрактний рівень опису даних Декларований рівень даних Базовий рівень даних Агрегований рівень даних <p>Структури зберігання даних</p> <ul style="list-style-type: none"> Концепція типу Квантування пам'яті Послідовні структури зберігання Спискові структури Розсіяні структури <p>Абстрактні структури даних та їхнє зображення структурами зберігання даних</p> <p>Масиви</p> <ul style="list-style-type: none"> Відображення масивів у структури зберігання <p>Таблиці</p> <ul style="list-style-type: none"> Таблиці з обчислювальними входами Таблиці з прямим доступом Перемішані таблиці Функція розстановки (функція хешування) <p>Рядки</p> <ul style="list-style-type: none"> Нормальні алгоритми Маркова Формальні граматики Операції над рядками Відображення рядків у структури зберігання <p>Динамічні структури</p>

	<ul style="list-style-type: none"> • Стеки • Відображення стеків у структури зберігання • Черги • Відображення черг у структури зберігання • Деки <p>Списки</p> <ul style="list-style-type: none"> • Відображення списків у структури зберігання <p>Деревовидні структури</p> <ul style="list-style-type: none"> • Основні визначення • Базові оператори над деревами • Відображення дерев у структури зберігання • Послідовні структури • Спискові структури зберігання • Застосування деревовидних структур • Древа сортувань <p>Сіткові структури</p> <p>Графи</p> <ul style="list-style-type: none"> • Основні поняття <p>Відображення графів у структури зберігання</p> <p>Множини</p>
	<p>Лабораторні</p> <ol style="list-style-type: none"> 1. Масив 2. Множина 3. Рядок 4. Нормальні алгоритми Маркова 5. Розрідженої матриці 6. Таблиці <ol style="list-style-type: none"> a. Таблиці з прямим доступом b. Перемішані таблиці 7. Динамічні структури <ol style="list-style-type: none"> a. Стек b. Черга c. Деревовидні структури. AVL дерево 8. Графи <p>Алгоритм Дейкстри</p>
Підсумковий контроль, форма	Диференційований залік
Схема курсу	Схема представлена в кінці документу
Пререквізити	Для вивчення курсу студенти потребують базових знань з дисциплін “Математика”, “Інформатика” за курс середньої школи та перший семестр курсу “Програмування”. Дисципліна “Алгоритми і структури даних” також тісно пов’язана з курсом “Програмування”, який вивчають у першому та другому семестрах.
Навчальні методи та техніки, які використовують під час викладання курсу	Лекції з мультимедійними презентаціями та з демонстрацією прийомів практичного використання середовища програмування; лабораторні заняття у вигляді проектування алгоритмів і програм, виконання практичних завдань; самостійне опрацювання навчальних матеріалів: підручників, конспектів лекцій. Обговорення теоретичного та практичного матеріалу в онлайн сервісах, формулювання творчих завдань для студентів, виконання яких готує до вивчення нового теоретичного матеріалу.
Необхідне обладнання	Для проведення лекцій: комп’ютер, проектор, доступ до мережі інтернет. Для проведення лабораторних та виконання завдань: комп’ютер, ОС Windows/Linux, доступ до інтернету, середовище програмування мовою C++ (Microsoft Visual Studio, Code Blocks тощо). Уся література, яку студенти не можуть знайти самостійно, буде надана викладачем виключно в освітніх цілях без права її передачі третім особам. Студенти заохочуються до використання також й іншої літератури та джерел, яких немає серед рекомендованих.
Критерії оцінювання (окремо для кожного виду	Оцінювання проводиться за 100-бальною шкалою. 100 балів нараховують за виконання лабораторних завдань. Лабораторні завдання є індивідуальними. Упродовж семестру студент виконує 10-11 лабораторних робіт, кожен з яких оцінюють в 10 балів.

<p>навчальної діяльності)</p>	<p>Для кожного завдання визначено термін виконання: зазвичай до закінчення навчального тижня. Вчасно виконані завдання оцінюють так (у відсотках від максимальної оцінки):</p> <ul style="list-style-type: none"> • 100% – умови завдання виконано повністю, алгоритми складено правильно, програма містить належні коментарі, роботу програми перевірено на достатньому наборі тестових даних, автор відповідає на всі запитання щодо використаних підходів, чітко інтерпретує отримані результати, немає ознак недоброчесності; • 80% – наведено логічно правильну послідовність розв’язування, алгоритми складено правильно, бракує окремих коментарів чи тестів, автор не досить повно пояснює використані підходи, немає ознак недоброчесності; • 60% – у правильній послідовності розв’язування допущено окремі помилки, які автор уміє виправити після зауваження викладача, бракує коментарів чи тестів, на запитання щодо використаних підходів автор відповідає з помилками, немає ознак недоброчесності; • 40% – у правильній послідовності розв’язування пропущено окремі етапи, завдання виконано частково, автор не розуміє недоліків поданої роботи, не вміє їх виправити, немає ознак недоброчесності; • 20% – завдання виконано частково, немає тестів, програма працює правильно для окремих наборів вхідних даних, автор не може самостійно інтерпретувати отримані результати, виправити помилки, немає ознак недоброчесності; • 0% – завдання не виконано, написана програма не відповідає умові, або ж виявлено ознаки недоброчесності: запозичення, фрагменти коду, дію яких автор пояснити не може, автор не володіє відповідним теоретичним матеріалом тощо; • можуть бути нараховані додаткові бали за повністю виконане завдання, яке містить кілька способів розв’язування, використовує особливо ефективний спосіб, демонструє креативність автора тощо. <p>Запізнення зменшує максимальну оцінку за завдання: кожного наступного після терміну виконання тижня оцінка зменшується удвічі.</p> <p>Відвідування занять є важливою складовою навчання. Очікується, що всі студенти відвідують усі лекції і лабораторні заняття курсу. Активність під час проведення лекцій і лабораторних заохочується балами. У будь-якому випадку студенти зобов’язані дотримуватися усіх строків визначених для виконання всіх видів робіт, передбачених курсом. Виконані роботи завантажують у відповідне хмарне сховище. Альтернативою відвідування лабораторних занять в університеті може бути дистанційна онлайн робота за розкладом проведення занять. Активність на лекціях і лабораторних враховують при оцінюванні відповідного лабораторного завдання.</p> <p>Академічна доброчесність: очікується, що роботи студентів будуть їхнім оригінальними дослідженнями чи міркуваннями. Відсутність посилань на використані джерела, фабрикування джерел, списування, втручання в роботу інших студентів, здавання чужих комп’ютерних програм як своїх становлять, але не обмежують, приклади можливої академічної недоброчесності. Виявлення ознак академічної недоброчесності в письмовій роботі студента є підставою для її не зарахування викладачем, незалежно від масштабів плагіату чи обману.</p>
<p>Опитування</p>	<p>Анкету-оцінку з метою оцінювання якості курсу буде надано після завершення курсу.</p>

Тиждень	Тема, план, короткі тези	Форма заняття	Тривалість, год
1	Вступ	Лекція	2
	Перевірка програм Настанови написання коду C++	самостійна робота	3
	Сортування	Лабораторна робота	2
2	Динамічні структури	Лекція	2
	Відображення стеків у структури зберігання	самостійна робота	4
	Стек	Лабораторна робота	2
3	Таблиці з обчислювальними входами	Лекція	2
	Таблиці з прямим доступом	самостійна робота	3
	Таблиці з прямим доступом	Лабораторна робота	2
4	Перемішані таблиці	Лекція	2
		самостійна робота	4
	Перемішані таблиці	Лабораторна робота	2
5	Множина	Лекція	2
		самостійна робота	3
	Множина	Лабораторна робота	2
6	Масиви	Лекція	2
		самостійна робота	4
	Розрідженої матриці	Лабораторна робота	2
7	Відображення масивів у структури зберігання	Лекція	2
		самостійна робота	4
	Розрідженої матриці	Лабораторна робота	2
8	Рядки	Лекція	2
		самостійна робота	3
	Розрідженої матриці	Лабораторна робота	2
9	Нормальні алгоритми Маркова	Лекція	2
	Формальні граматики	самостійна робота	3
	Нормальні алгоритми Маркова	Лабораторна робота	2
10	Операції над рядками	Лекція	2
	Відображення рядків у структури зберігання	самостійна робота	4
	Нормальні алгоритми Маркова	Лабораторна робота	2
11	Динамічні структури	Лекція	2
	Черги	самостійна робота	3
	Відображення черг у структури зберігання		
	Черга	Лабораторна робота	2

12	Динамічні структури Відображення черг у структури зберігання Деки	Лекція самостійна робота	2 3
	Черга	Лабораторна робота	2
13	Деревовидні структури Основні визначення Базові оператори над деревами Відображення дерев у структури зберігання	Лекція самостійна робота	2 4
	AVL дерево	Лабораторна робота	2
14	Деревовидні структури Послідовні структур Спискові структури зберігання и	Лекція самостійна робота	2 4
	AVL дерево	Лабораторна робота	2
15	Графи Основні поняття Відображення графів у структури зберігання	Лекція самостійна робота	2 3
	Графи-Алгоритм Дейкстри.	Лабораторна робота	2
16	Багаторівневе відображення даних	Лекція самостійна робота	2 4
	Графи-Алгоритм Дейкстри	Лабораторна робота	2