

**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 та  
інформатики  
Львівського національного університету  
імені Івана Франка  
(протокол № 1 від 29 серпня 2023 р.)



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

**Силабус навчальної дисципліни**  
**«Алгоритми і структури даних»,**  
**що викладається в межах ОПІ Інформатика**  
**першого (бакалаврського) рівня вищої освіти для здобувачів з**  
**спеціальності 122 – Комп'ютерні науки**

Львів 2023 р.

<b>Назва дисципліни</b>	Алгоритми і структури даних
<b>Адреса викладання дисципліни</b>	Львівський національний університет імені Івана Франка, вул. Університетська 1, м. Львів, Україна, 79000
<b>Факультет та кафедра, за якою закріплена дисципліна</b>	Факультет прикладної математики та інформатики, кафедра програмування
<b>Галузь знань, шифр та назва спеціальності</b>	12 – інформаційні технології 122 – Комп'ютерні науки
<b>Викладачі дисципліни</b>	Літинський Святослав Володимирович, к. ф.-м. н., доцент, доцент кафедри програмування
<b>Контактна інформація викладачів</b>	Електронна пошта: <a href="mailto:svyatoslav.litynskyi@lnu.edu.ua">svyatoslav.litynskyi@lnu.edu.ua</a> веб-сторінка: <a href="https://ami.lnu.edu.ua/employee/litynskyi">https://ami.lnu.edu.ua/employee/litynskyi</a>
<b>Консультації з питань навчання по дисципліні відбуваються</b>	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.
<b>Сторінка курсу</b>	<a href="https://ami.lnu.edu.ua/course/algorithms-and-data-structures-informatics">https://ami.lnu.edu.ua/course/algorithms-and-data-structures-informatics</a>
<b>Інформація про дисципліну</b>	Курс “Алгоритми і структури даних” є нормативною дисципліною зі спеціальності 122 – Комп'ютерні науки для освітньої програми Інформатика, яка викладається в 2-му семестрі в обсязі 4 кредитів (за Європейською Кредитно-Трансферною Системою ECTS).
<b>Коротка анотація дисципліни</b>	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.
<b>Мета та цілі дисципліни</b>	Метою нормативної дисципліни «Алгоритми і структури даних» є навчити студента: <ul style="list-style-type: none"> <li>● базовим структурам даних та їхнім перевагам і недолікам;</li> <li>● реалізувати їх мовою C++;</li> <li>● виконувати часову та ємнісну оцінку складності алгоритмів з різними структурами даних;</li> <li>● застосовувати основні структури даних у прикладних задачах.</li> </ul>
<b>Література для вивчення дисципліни</b>	<i>Основна література</i> 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. <i>Додаткова література</i> 3. Ярошко С. А. Методи розробки алгоритмів. Програмування мовою C++: навч. посібник / С.А. Ярошко, О.С. Ярошко – Львів: ЛНУ імені Івана Франка, 2022. – 248 с. – ISBN 978-617-10-0718-5. – <a href="https://lnuittutor.github.io/">https://lnuittutor.github.io/</a>
<b>Обсяг курсу</b>	4 кредити ЄКТС – 120 годин. З них 32 годин лекцій, 32 години лабораторних занять та 56 годин самостійної роботи
<b>Очікувані результати навчання</b>	Після завершення цього курсу студент буде: <p><i>знати</i></p> <ul style="list-style-type: none"> <li>● про існування, способи реалізації та характеристики часової та ємнісної складності різних структур даних.</li> </ul> <p><i>вміти</i></p> <ul style="list-style-type: none"> <li>● Написати самостійно та використати існуючу бібліотеку з різними структурами даних.</li> </ul>
<b>Компетентності</b>	<i>Загальні:</i>

	<p>ЗК2. Здатність застосовувати знання у практичних ситуаціях.</p> <p>ЗК3. Знання та розуміння предметної області та розуміння професійної діяльності.</p> <p><i>Спеціальні компетентності:</i></p> <p>СК8. Здатність проектувати 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.</p> <p>СК10. 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.</p> <p>СК17. Здатність розв'язувати прикладні задачі на основі інтегрованого знання про основні методи інформатики та інформаційні технології, використовувати комп'ютер і технології зв'язку, представляти дані у зрозумілій для всіх формі, яка проявляється у прагненні, здатності і готовності до ефективного застосування сучасних засобів інформаційних та комп'ютерних технологій для розв'язання завдань у професійній діяльності, усвідомлюючи при цьому значущість предмету і результату діяльності.</p>
<p><b>Програмні результати навчання</b></p>	<p>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.</p> <p>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.</p> <p>ПР18. Розуміти принципи багаторівневого відображення даних від абстрактних структур даних до структур фізичної пам'яті комп'ютера, вміти створювати програмну реалізацію структур даних, будувати ефективні алгоритми їхнього використання.</p>
<p><b>Ключові слова</b></p>	<p>Алгоритм, тип даних, структура даних, часова складність, просторова складність, стек, дек, дерево, граф, хеш таблиця, список.</p>
<p><b>Формат курсу</b></p>	<p>Очний.</p>
<p><b>Теми</b></p>	<p><b>Лекції</b></p> <p>Вступ</p> <p>Багаторівневе відображення даних</p> <ul style="list-style-type: none"> <li>• Змістовний рівень опису даних</li> <li>• Абстрактний рівень опису даних</li> <li>• Декларований рівень даних</li> <li>• Базовий рівень даних</li> <li>• Агрегований рівень даних</li> </ul> <p>Структури зберігання даних</p> <ul style="list-style-type: none"> <li>• Концепція типу</li> <li>• Квантування пам'яті</li> <li>• Послідовні структури зберігання</li> <li>• Спискові структури</li> <li>• Розсіяні структури</li> </ul> <p>Абстрактні структури даних та їхнє зображення структурами зберігання даних</p>

	<p>Масиви</p> <ul style="list-style-type: none"> <li>Відображення масивів у структури зберігання</li> </ul> <p>Таблиці</p> <ul style="list-style-type: none"> <li>Таблиці з обчислювальними входами</li> <li>Таблиці з прямим доступом</li> <li>Перемішані таблиці</li> <li>Функція розстановки (функція хешування)</li> </ul> <p>Рядки</p> <ul style="list-style-type: none"> <li>Нормальні алгоритми Маркова</li> <li>Формальні граматики</li> <li>Операції над рядками</li> <li>Відображення рядків у структури зберігання</li> </ul> <p>Динамічні структури</p> <ul style="list-style-type: none"> <li>Стеки</li> <li>Відображення стеків у структури зберігання</li> <li>Черги</li> <li>Відображення черг у структури зберігання</li> <li>Деки</li> </ul> <p>Списки</p> <ul style="list-style-type: none"> <li>Відображення списків у структури зберігання</li> </ul> <p>Деревовидні структури</p> <ul style="list-style-type: none"> <li>Основні визначення</li> <li>Базові оператори над деревами</li> <li>Відображення дерев у структури зберігання</li> <li>Послідовні структури</li> <li>Спискові структури зберігання</li> <li>Застосування деревовидних структур</li> <li>Дерева сортувань</li> </ul> <p>Сіткові структури</p> <p>Графи</p> <ul style="list-style-type: none"> <li>Основні поняття</li> <li>Відображення графів у структури зберігання</li> </ul> <p>Множини</p>			
<b>Підсумковий контроль, форма</b>	диференційований залік			
<b>Схема курсу.</b>	Тиждень	Тема, план, короткі тези	Форма заняття	Тривалість, год
	1	Вступ Перевірка програм Настанови написання коду C++ Сортування	Лекція самостійна робота Лабораторна робота	2 3 2
	2	Динамічні структури Відображення стеків у структури зберігання Стек	Лекція самостійна робота Лабораторна робота	2 4 2
	3	Таблиці з обчислювальними входами Таблиці з прямим доступом Таблиці з прямим доступом	Лекція самостійна робота Лабораторна робота	2 3 2
	4	Перемішані таблиці	Лекція самостійна робота	2 4

	Перемішані таблиці	Лабораторна робота	2
5	Множина	Лекція самостійна робота	2 3
	Множина	Лабораторна робота	2
6	Масиви	Лекція самостійна робота	2 4
	Розрідженої матриці	Лабораторна робота	2
7	Відображення масивів у структури зберігання	Лекція самостійна робота	2 4
	Розрідженої матриці	Лабораторна робота	2
8	Рядки	Лекція самостійна робота	2 3
	Розрідженої матриці	Лабораторна робота	2
9	Нормальні алгоритми Маркова Формальні граматики	Лекція самостійна робота	2 3
	Нормальні алгоритми Маркова	Лабораторна робота	2
10	Операції над рядками Відображення рядків у структури зберігання	Лекція самостійна робота	2 4
	Нормальні алгоритми Маркова	Лабораторна робота	2
11	Динамічні структури Черги Відображення черг у структури зберігання	Лекція самостійна робота	2 3
	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 структури Спискові структури зберігання	Лекція самостійна робота	2 4
	AVL дерево	Лабораторна робота	2
15	Графи Основні поняття	Лекція самостійна робота	2 3

		Відображення графів у структури зберігання		
		Графи-Алгоритм Дейкстри.	Лабораторна робота	2
	16	Багаторівневе відображення даних	Лекція самостійна робота	2 4
		Графи-Алгоритм Дейкстри	Лабораторна робота	2
<b>Пререквізити</b>	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.			
<b>Навчальні методи та техніки, які використовують під час викладання курсу</b>	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.			
<b>Необхідне обладнання</b>	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.			
<b>Критерії оцінювання (окремо для кожного виду навчальної діяльності)</b>	<p><b>Оцінювання</b> проводиться за 100-бальною шкалою. 100 балів нараховують за виконання лабораторних завдань. Лабораторні завдання є індивідуальними. Упродовж семестру студент виконує 10 лабораторних робіт, кожен з яких оцінюють в 10 балів.</p> <p><b>Відвідування занять</b> 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><b>Академічна доброчесність:</b> 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. Жодні форми порушення академічної доброчесності не толеруються.</p>			
<b>Опитування</b>	Анкету-оцінку з метою оцінювання якості курсу буде надано після завершення курсу.			