

**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 31 серпня 2025 р.)

---

Завідувач кафедри Ярошко С.А.

**Силабус з навчальної дисципліни**  
**“Вступ до програмування”,**  
**що викладається в межах ОПП “Системний аналіз і управління.**  
**Інтелектуальний аналіз даних”**  
**першого (бакалаврського) рівня вищої освіти для здобувачів з**  
**спеціальності 124 – системний аналіз**

Львів 2025 р.

<b>Назва дисципліни</b>	Вступ до програмування
<b>Адреса викладання дисципліни</b>	Головний корпус ЛНУ ім. І. Франка м. Львів, вул. Університетська 1
<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 124 – системний аналіз
<b>Викладачі дисципліни</b>	Гошко Богдан Мирославович, доцент кафедри програмування
<b>Контактна інформація викладача</b>	<a href="mailto:bohdan.hoshko@lnu.edu.ua">bohdan.hoshko@lnu.edu.ua</a> ; <a href="https://ami.lnu.edu.ua/employee/hoshko-b-m">https://ami.lnu.edu.ua/employee/hoshko-b-m</a> Головний корпус ЛНУ ім. І. Франка, каб. 263. м. Львів, вул. Університетська, 1
<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 Zoom чи Microsoft Teams. Для погодження часу он-лайн консультацій слід писати на електронну пошту викладача.
<b>Сторінка курсу</b>	<a href="http://ami.lnu.edu.ua/course/vstup-do-prohramuvannya-ssystemnyj-analiz">http://ami.lnu.edu.ua/course/vstup-do-prohramuvannya-ssystemnyj-analiz</a>
<b>Інформація про дисципліну</b>	Курс є нормативною дисципліною зі спеціальності 124 Системний аналіз для освітньої програми Системний аналіз і управління. Інтелектуальний аналіз даних, яку викладають у другому семестрі в обсязі 5 кредитів (за Європейською кредитно-трансферною системою ECTS)
<b>Коротка анотація дисципліни</b>	Фокус уваги курсу спрямовано на те, щоб навчити вчорашнього випускника писати досить складні ефективні комп'ютерні програми. Одним з предметів вивчення і, водночас, головним засобом навчання є мова програмування C++. Перший семестр навчання закладає базу: дає основи алгоритмізації та вступ до об'єктно-орієнтованого програмування
<b>Мета та цілі дисципліни</b>	Метою нормативної дисципліни «Програмування» є навчити студента: <ul style="list-style-type: none"> <li>● складати алгоритми, оцінювати їхню обчислювальну ефективність, перевіряти правильність за допомогою сукупності тестів;</li> <li>● вибирати відповідні засоби мови програмування і застосовувати їх належним чином для запису своїх алгоритмів розв'язування прикладних задач;</li> <li>● застосовувати процедурну методологію побудови комп'ютерних програм;</li> <li>● застосовувати об'єктно-орієнтований підхід до проектування і написання програм,</li> </ul>
<b>Література для вивчення дисципліни</b>	<ol style="list-style-type: none"> <li>1. Бублик В.В. Об'єктно-орієнтоване програмування: [Підручник] / В.В. Бублик. – К.: ІТ-книга, 2015. – 624 с.</li> <li>2. Дудзяний І.М. Програмування мовою C++. Частина 1: Парадигма процедурного програмування: навчальний посібник / І.М. Дудзяний. – Львів: ЛНУ імені Івана Франка, 2013. – 468 с.</li> <li>3. Stroustrup, Bjarne Programming: Principles and Practice Using C++/ Bjarne Stroustrup, Addison-Wesley. Second edition. – 2013. -</li> <li>4. Stroustrup, Bjarne A Tour of C++ 2nd Edition / Bjarne Stroustrup. Addison-Wesley. Second edition. – 2018. -181с.</li> <li>5. Ярошко С.А. Методи розробки алгоритмів. Навчальний посібник / С.А. Ярошко – Львів, ЛНУ імені Івана Франка, 2019. [Електронне видання]</li> </ol>

<b>Обсяг курсу</b>	Загальний обсяг: 120 годин. Аудиторних занять: 64 год., з них 32 год. лекційних та 32 годин лабораторних робіт. Самостійної роботи: 56 год.
<b>Очікувані результати навчання</b>	<p>У результаті вивчення даного курсу студент буде:</p> <p><b>знати:</b></p> <ul style="list-style-type: none"> <li>● основні концепції мови C++,</li> <li>● парадигми програмування, які підтримує мова C++: процедурне, об'єктно-орієнтоване та узагальнене програмування,</li> <li>● типи даних, вбудовані та користувача,</li> <li>● методи сортування і пошуку даних</li> </ul> <p><b>вміти:</b></p> <ul style="list-style-type: none"> <li>● реалізовувати алгоритми за допомогою програм,</li> <li>● створювати функції для обробки даних,</li> <li>● утворювати нові типи даних за допомогою класів, структур та взаємозв'язків між типами.</li> </ul>
<b>Ключові слова</b>	C++, тип даних, оператор, клас, структура, функція, метод класу
<b>Формат курсу</b>	Очний: проведення лекцій, лабораторних робіт та консультацій в приміщеннях університету, а в умовах карантину – онлайнний на платформі Microsoft Teams
<b>Теми</b>	<p>Тема 1. Три парадигми програмування, які підтримує C++.</p> <p>Тема 2. Система типів.</p> <p>Тема 3. Інструкції циклу for, while, do while.</p> <p>Тема 4. Дійсні типи. Перетворення типів. Математичні функції (cmath).</p> <p>Тема 5. Загальна структура оголошення, typedef. Структуровані типи даних C++.</p> <p>Тема 6. Вказівники, арифметика вказівників, застосування до масивів.</p> <p>Тема 7. Функції в C++: прототип, визначення, виклик..</p> <p>Тема 8. Компоненти і протоколи корпоративної мережі.</p> <p>Тема 9. Структуровані типи даних C++: структури, бітові поля, об'єднання.</p> <p>Тема 10. Моделювання структури даних зв'язний список.</p> <p>Тема 11. Основні алгоритми впорядкування масивів.</p> <p>Тема 12. Функції для опрацювання C-рядків.</p> <p>Тема 13. Моделювання структури даних двійкове дерево</p> <p>Тема 14. Оголошення та використання просторів імен</p> <p>Тема 15. Оголошення класу і визначення методів класу.</p> <p>Тема 16. Використання дружніх функцій</p>
<b>Підсумковий контроль, форма</b>	Комбінований іспит
<b>Пререквізити</b>	Для вивчення курсу студенти потребують базових знань з дисциплін “Математика”, “Інформатика” за курс середньої школи. Одночасно з вивченням програмування студенти проходять навчальну обчислювальну практику, впродовж якої виконують завдання з програмування, поглиблюють набуті в курсі знання та удосконалюють навички. Дисципліна “Вступ до програмування” також тісно пов'язана з курсом “Алгоритми обчислювальних процесів”, який вивчають у другому семестрі.
<b>Навчальні методи та техніки, які будуть використовуватися під час</b>	Лекції з мультимедійними презентаціями та з демонстрацією способів практичного використання середовища програмування; лабораторні заняття у вигляді проектування алгоритмів і програм, виконання практичних завдань, у тому числі у команді з 3-4 осіб; самостійне опрацювання навчальних матеріалів: підручників, конспектів лекцій, готових програм мовою C++, додаткових навчальних посібників, розміщених у хмарному сховищі (Moodle, Microsoft Teams, Google Classroom). Обговорення теоретичного та

<b>викладання курсу</b>	практичного матеріалу в онлайн сервісах, формулювання творчих завдань для студентів, виконання яких готує до вивчення нового теоретичного матеріалу
<b>Необхідне обладнання</b>	<p>Для проведення лекцій: комп'ютер, проектор, доступ до мережі інтернет.</p> <p>Для проведення лабораторних та виконання завдань: комп'ютер, ОС Windows/Linux, доступ до інтернету, середовище програмування мовою C++ (Microsoft Visual Studio, Code Blocks тощо).</p> <p>Уся література, яку студенти не зможуть 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.</p>
<b>Критерії оцінювання (окремо для кожного виду навчальної діяльності)</b>	<p><b>Оцінювання</b> проводиться за 100-бальною шкалою. 50 балів нараховують за виконання лабораторних завдань і контрольних робіт, ще 50 балів – за виконання екзаменаційного завдання. Лабораторні завдання можуть бути індивідуальні та командні. Упродовж семестру студент виконує не менше 10 лабораторних робіт, кожна з яких оцінюють 3-5 балів залежно від складності. Оцінка за екзаменаційне завдання може бути поділена на дві частини: 20 балів за засвоєння теоретичного матеріалу, виставлені після опитувань упродовж семестру (у формі тестувань, колоквиумів тощо) та 30 за написання комп'ютерної програми (декількох програм).</p> <p><b>Відвідання занять</b> є важливою частиною навчання. Очікується, що всі студенти відвідають 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 усіх видів письмових 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 ураховують при оцінюванні відповідного лабораторного завдання.</p> <p><b>Академічна доброчесність:</b> Очікується, що роботи студентів будуть їх оригінальними дослідженнями чи міркуваннями. Відсутність посилань на використані джерела, фабрикування джерел, списування, втручання в роботу інших студентів становлять, але не обмежують, приклади можливої академічної недоброчесності. Виявлення ознак академічної недоброчесності в написанні програм є підставою для її незарахування викладачем, незалежно від масштабів плагіату чи обману.</p> <p><b>Політика виставлення балів.</b> Враховуються бали набрані при поточному тестуванні, самостійній роботі та бали підсумкового тестування. При цьому обов'язково враховуються присутність на заняттях та активність студента під час практичного заняття; недопустимість пропусків та запізнь на заняття; користування мобільним телефоном, планшетом чи іншими мобільними пристроями під час заняття в цілях не пов'язаних з навчанням; списування та плагіат; несвоєчасне виконання поставленого завдання і т. ін.</p> <p><b>Жодні форми порушення академічної доброчесності не толеруються.</b></p>
<b>Питання до іспиту</b>	<p>Які типи в C++ використовують для зберігання значень цілого типу? Наведіть все можливі приклади зображення цілого. Які типи в C++ використовують для зберігання значень дійсного типу? Наведіть все можливі приклади зображення дійсного. Які типи в C++ використовують для зберігання літер? Наведіть всеможливі приклади зображення літер. Вкажіть оператори C++, які застосовують до значень цілих типів (у порядку спадання пріоритету). Вкажіть оператори C++, які застосовують до значень дійсних типів (у порядку спадання пріоритету). Вкажіть оператори присвоєння C++, поясніть алгоритм їхнього виконання. Який тип C++ використовують для зберігання логічних значень? Як зображають логічні значення? Які оператори до них застосовують? Наведіть</p>

	<p>приклад вкороченої умовної інструкції мови C++. Поясніть алгоритм її виконання. Наведіть приклад повної умовної інструкції мови C++. Поясніть алгоритм її виконання. Наведіть приклад інструкції switch мови C++. Поясніть алгоритм її виконання. Наведіть приклад інструкції for мови C++. Поясніть алгоритм її виконання. Наведіть приклад інструкції while мови C++. Поясніть алгоритм її виконання. Наведіть приклад інструкції do мови C++. Поясніть алгоритм її виконання. Наведіть приклад використання оператора ? : мови C++. Поясніть алгоритм його виконання. Інструкція переходу мови C++: синтаксис, алгоритм виконання, випадки використання. Інструкція break мови C++: синтаксис, алгоритм виконання, випадки використання (перелічіть усі). Інструкція return мови C++: синтаксис, алгоритм виконання, випадки використання (перелічіть усі). Що таке typedef у мові C++? Як його можна використати? Оголошення одновимірного масиву в C++. Ініціалізація. Звертання до елементів масиву. Оголошення двохвимірного масиву в C++. Ініціалізація. Звертання до елементів масиву. Опишіть функцію мовою C++ для виведення на друк одновимірного числового масиву. Наведіть фрагмент коду для виведення на друк двохвимірного числового масиву. Що таке вказівник? Як його оголосити? Вкажіть всі способи ініціалізації вказівника. Як створити динамічну змінну простого типу? Як задати їй значення? Як її знищити? Як створити одновимірний динамічний масив? Як його знищити? Як ввести (вивести) значення одновимірного динамічного масиву? Як створити двохвимірний динамічний масив? Які оператори можна застосувати до вказівника? Поясніть. Що потрібно для того, аби використати функцію в C++-програмі? Що таке прототип функції? Навіщо він потрібен? Наведіть приклад. Які є два основні різновиди функцій у C++? Яка між ними різниця? Які є способи передавання аргументів функціям? Поясніть синтаксис і семантику передавання аргументів за значенням. Поясніть синтаксис і семантику передавання аргументів за адресою. Поясніть синтаксис і семантику передавання аргументів через посилання. Використання посилань у параметрах функції. Переваги та недоліки. Як виконують виклик функції у C++? Наведіть приклади. Що таке локальні змінні функції? (Оголошення, використання, видимість, час існування, приклади). Алгоритм відшукування найбільшого з трьох заданих чисел. Алгоритм обчислення суми значень числового масиву. Алгоритм відшукування найбільшого значення числового масиву. Алгоритм переведення цілого числа до нової системи числення. Алгоритм перевірки впорядкованості числового масиву. Алгоритм вставки числа у впорядкований масив. Алгоритм злиття двох впорядкованих масивів. Алгоритм впорядкування масиву методом обміну (бульбашки).</p>
<p><b>Опитування</b></p>	<p>Анкету-оцінку з метою оцінювання якості курсу буде надано по завершенню курсу.</p>