

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

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



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

Силабус з навчальної дисципліни
“Навчальна (комп’ютерна) практика”,
що викладається в межах ОПП “Середня освіта (інформатика)”
першого (бакалаврського) рівня вищої освіти
для здобувачів з предметної спеціальності А4.09 Середня освіта
(Інформатика)
галузі знань А Освіта/Педагогіка

Львів 2025 р.

Назва дисципліни	Навчальна (комп'ютерна) практика
Адреса викладання дисципліни	Головний корпус ЛНУ ім. І. Франка м. Львів, вул. Університетська 1
Факультет та кафедра, за якою закріплена дисципліна	Факультет прикладної математики та інформатики Кафедра інформаційних систем Кафедра програмування
Галузь знань, шифр та назва спеціальності	Галузь знань: А Освіта / Педагогіка, Спеціальність: А4 Середня освіта Спеціалізація: А4.09 Середня освіта (Інформатика)
Викладачі дисципліни	Ярошко Сергій Адамович, завідувач кафедри програмування Ярошко Оксана Сергіївна, асистент кафедри інформаційних систем;
Контактна інформація викладачів	https://ami.lnu.edu.ua/employee/yaroshko https://ami.lnu.edu.ua/employee/yaroshko-o-s Головний корпус ЛНУ ім. І. Франка, каб. 260 м. Львів, вул. Університетська, 1
Консультації з питань навчання по дисципліні відбуваються	Консультації в день проведення практичних занять (за попередньою домовленістю). Також можливі онлайн консультації у Zoom, MS Teams.
Сторінка курсу	https://ami.lnu.edu.ua/course/navchalna-komp-iutema-praktyka-kr
Інформація про дисципліну	Дисципліна “Навчальна (комп'ютерна) практика” є нормативною дисципліною з спеціальності А4 – Середня освіта (інформатика) для освітньої програми “Середня освіта (інформатика)”, яка викладається в 1-му і 2-му семестрах в обсязі 4,5 кредитів (за Європейською Кредитно-Трансферною Системою ECTS).
Коротка анотація дисципліни	“Навчальна (комп'ютерна) практика” посідає важливе місце у процесі фахової підготовки спеціаліста з даної спеціальності, оскільки під час практики студенти вчаться використовувати теоретичні знання, здобуті з курсу “Програмування”, для розробки програм.
Мета та цілі дисципліни	Мета і цілі вивчення нормативної дисципліни “Навчальна (комп'ютерна) практика” – поглиблення і закріплення здобутих теоретичних знань з програмування мовою С++ з використанням об'єктно-орієнтованого програмування і стандартної бібліотеки шаблонів; розвиток логічного мислення; набуття професійних навиків з розробки програмного забезпечення, орієнтованого на розв'язання математичних задач.
Література для вивчення дисципліни	<i>Основна література</i> 1. Ярошко С. А. Методи розробки алгоритмів. Програмування мовою С++: навч. посібник / С.А. Ярошко, О.С. Ярошко – Львів: ЛНУ імені Івана Франка, 2022. – 248 с. – https://lnuittutor.github.io/ 2. Бублик В.В. Об'єктно-орієнтоване програмування: [Підручник] / В.В. Бублик. – К.: ІТ-книга, 2015. – 624 с. 3. Stephen Prata С++ Primer Plus 6th Edition (Developer's Library) In 2 Volumes – Addison-Wesley Professional, 2011. – 1440 p. 4. Дудзяний І.М. Програмування мовою С++. Частина 1: Парадигма процедурного програмування: навчальний посібник / І.М. Дудзяний. – Львів: ЛНУ імені Івана Франка, 2013. – 468 с. 5. Microsoft Learn: С++ language documentation. – Електронний ресурс. Режим доступу: https://learn.microsoft.com/en-us/cpp/cpp/?view=msvc-170 6. https://git-scm.com/book/uk/v2 <i>Додаткова література</i>

	<p>7. Bruce Eckel Thinking in C++, Vol. 1: Introduction to Standard C++, 2nd Edition – Prentice Hall, 2000. – 814 p.</p> <p>8. Bruce Eckel Thinking in C++, Vol. 2: Practical Programming, 2nd Edition – Prentice Hall, 2003. – 832 p.</p> <p>9. Scott Meyers Effective Modern C++ – O’Reilly Media, 2015. – 316 p.</p> <p>10. Nicolai M. Josuttis The C++ Standard Library. A tutorial and Reference. Second Edition – Addison-Wesley, 2012. – 1162 p.</p> <p>11. http://www.cplusplus.com/</p> <p>12. https://www.geeksforgeeks.org/c-plus-plus/</p> <p>13. https://metanit.com/cpp/</p> <p>14. https://www.codeproject.com/</p>
Обсяг курсу	Загальний обсяг: 135 годин. З них 96 год. аудиторних занять (практичних робіт) і 39 год. самостійної роботи.
Очікувані результати навчання	<p>Після завершення цього курсу студент буде :</p> <p>Знати:</p> <ul style="list-style-type: none"> - основні конструкції мови програмування C++; - основи процедурного програмування; - основи об’єктно-орієнтованого програмування; - стандартну бібліотеку шаблонів; - правила розробки прикладних програм. <p>Вміти:</p> <ul style="list-style-type: none"> - самостійно розробляти прості алгоритми для вирішення задачі; - використовувати програмне середовище MS Visual Studio; - використовувати набуті знання та навички для розроблення консольних застосувань на мові C++ з використанням методів об’єктно-орієнтованого програмування; - проектувати, розробляти, налагоджувати та тестувати програми.
Компетентності	<p>Курс забезпечує набуття таких компетентностей:</p> <p>ЗК2- Знання й розуміння предметної області та професійної діяльності.,</p> <p>ЗК6- Здатність до міжособистісної взаємодії та роботи у команді у сфері професійної діяльності, спілкування з представниками інших професійних груп різного рівня,</p> <p>ПК2- Володіння методами інформаційного моделювання; здатність реалізовувати інформаційну модель засобами інформаційно-комунікаційних технологій; проводити комп’ютерний експеримент, інтерпретувати, аналізувати та узагальнювати його результати,</p> <p>ПК8 Здатність до цифрового подання та обробки текстової, числової, графічної, звукової та відеоінформації.</p>
Програмні результати навчання	<p>та програмних результатів навчання:</p> <p>ПРН7- Демонструє знання основ фундаментальних і прикладних наук інформатики та програмування, оперує базовими категоріями та поняттями предметної області спеціальності.,</p> <p>ПРН 13- Визначає структуру предметної галузі інформатики, її місце в системі наук, пояснює перспективи розвитку інформатики та інформаційних технологій, їхнє суспільне значення,</p> <p>ПРН14- Знає та розуміє фізичні, логічні та математичні основи інформаційних технологій; пояснює та застосовує способи двійкового кодування текстової, числової, графічної, звукової та відеоінформації.</p>
Ключові слова	програмування, ООП, C++, стандартна бібліотека шаблонів.
Формат курсу	Очний,

Теми	<p style="text-align: center;">1 семестр</p> <p>Вступ. Ознайомлення з головними завданнями практики. <i>Індивідуальні завдання</i></p> <ol style="list-style-type: none"> 1. Реалізація обчислень за математичними формулами (послідовніалгоритми), наочне введення, виведення. – 1-й тиждень, 4 бали. 2. Програмна реалізація алгоритмів з розгалуженнями. – 2-й тиждень, 3 бали. 3. Побудова арифметичних й ітераційних циклів (задачі цілочисловоїарифметики, опрацювання послідовностей). – 3-й тиждень, 4 бали. 4. Побудова, трансформація одно- та багатовимірних масивів, у томучислі динамічних. – 4-й тиждень, 4 бали. 5. Опрацювання рядків. – 5-й тиждень, 3 бали. 6. Визначення та використання простих функцій. Рекурсія. – 6-й – 7-й тижні, 5 балів. 7. Визначення та використання функцій вищого порядку. – 8-й тиждень, 4 бали. 8. Визначення та використання простих структур, перевизначенняоператорів. – 9-й – 10-й тижні, 5 балів. 9. Використання файлів різних форматів для зберігання та завантаження даних. – 11-й тиждень, 4 бали. 10. Опрацювання виняткових ситуацій. – 12-й тиждень, 4 бали. <p><i>Командні завдання</i></p> <ol style="list-style-type: none"> 11. Моделювання зв'язних структур: одно- та двозв'язних списків, бінарних дерев, дерев пошуку. – 13-й – 14-й тижні, 5 балів. 12. Створення найпростіших класів і об'єктів. – 15-й – 16-й тижні, 5 балів <p style="text-align: center;">2 семестр</p> <p><i>Командні завдання</i></p> <ol style="list-style-type: none"> 13. Розробка ієрархії класів. Просте наслідування класів, створенняполіморфного масиву екземплярів. – 1-й – 2-й тижні, 5 балів. 14. Побудова класів-комполітів. – 3-й – 4-й тижні, 5 балів. 15. Розробка власного класу-контейнера з багатими функціональними можливостями. – 5-й – 6-й тижні, 5 балів. <p><i>Індивідуальні завдання</i></p> <ol style="list-style-type: none"> 16. Випробування множинного, закритого, захищеного наслідування. – 7-й тиждень, 5 балів. 17. Розробка та використання шаблону функції. – 8-й тиждень, 5 балів. 18. Розробка та використання шаблону класу. – 9-й тиждень, 5 балів. 19. Використання послідовних контейнерів і алгоритмів бібліотеки STL для створення/опрацювання колекції об'єктів. – 10-й – 11-й тижні, 5 балів. 20. Використання асоціативних контейнерів і алгоритмів бібліотеки STL для створення/опрацювання колекції об'єктів. – 12-й – 13-й тижні, 5 балів. 21. Оголошення та використання класів характеристик для налаштуванняшаблону функції/класу. – 14-й тиждень, 5 балів. <p>Серіалізація/десеріалізація об'єктів. – 15-й – 16-й тижні, 5 балів.</p>
Підсумковий контроль, форма	Диференційований залік у кінці другого семестру

Пререквізити	Для проходження практики студенти потребують базових знань з курсів «Програмування»; «Алгебра та геометрія»; «Дискретна математика»; «Алгоритми і структури даних», достатніх для розроблення застосунків мовою C++.
Навчальні методи та техніки, які будуть використовуватися під час викладання курсу	Навчальна (обчислювальна) практика покликана виявити та розвинути індивідуальні здібності здобувача освіти щодо створення програм мовою C++, тому вона передбачає здебільшого самостійне виконання завдань. Студент пише програми, перевіряє їхню правильність, оформляє звіт про виконане завдання і захищає його. Керівник практики надає консультації щодо формулювання завдань, допомагає з правильним налаштуванням перших програмних проектів, сховищ. Тренує зі здобувачами правильну послідовність дій щодо покрокового виконання програм, створення тестів тощо. Керівник організовує колективні обговорення командних завдань, допомагає з розподілом обов'язків і презентацією отриманих результатів.
Необхідне обладнання	Комп'ютер, ОС Windows/Linux, доступ до інтернету, середовище програмування мовою C++ (Microsoft Visual Studio, Code Blocks тощо). 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.
Критерії оцінювання (окремо для кожного виду навчальної діяльності)	<p>Оцінювання виконують за 100-бальною шкалою: 50 балів за виконані завдання у першому семестрі, 50 балів – у другому. Звіт про практику можна оформити як окремий документ, або як належним чином оформлене сховище (сховища) на GitHub. Опис кожного завдання повинен містити умову, текст програми з коментарями, аналіз отриманих результатів, модульні тести.</p> <p>Критерії оцінювання індивідуальних завдань:</p> <p>100% балів – студент повністю виконав умови завдання, хід розв'язування оформлено належним чином, нема ознак недоброчесності.</p> <p>80% балів – студент повністю виконав умови завдання, хід розв'язування оформлено належним чином, розв'язок містить незначні помилки, нема ознак недоброчесності.</p> <p>60-50% балів – студент виконав завдання з помилками, або ж не повністю викладено хід розв'язування завдання, нема ознак недоброчесності;</p> <p>20% балів – студент виконав завдання частково з грубими помилками, які самостійно не може виправити, демонструє незнання матеріалу, нема ознак недоброчесності.</p> <p>0 балів – студент не виконав завдання або ж у роботі виявлені ознаки недоброчесності.</p> <p>Додаткові бали будуть зараховані учасникам Всеукраїнської студентської олімпіади з програмування (відповідно до кількості розв'язаних задач).</p> <p>Відвідання занять є важливою складовою навчання. Очікується, що всі студенти відвідають усі заняття практики. Активність під час занять заохочується балами. У будь-якому випадку студенти зобов'язані дотримуватися усіх строків виконання завдань практики. Виконані роботи завантажують у відповідне хмарне сховище. Якщо заняття за наказом ректора відбуваються дистанційно, студенти повинні долучатися до онлайн-нарад з увімкненими відеокамерами.</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 незарахування викладачем, незалежно від масштабів плагіату чи обману.</p>
Опитування	Анкету-оцінку з метою оцінювання якості курсу буде надано по завершенню курсу.