

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Львівський національний університет імені Івана Франка
Факультет електроніки та комп'ютерних технологій
Кафедра радіофізики та комп'ютерних технологій

Затверджено

На засіданні
кафедри радіофізики та комп'ютерних
технологій
факультету електроніки та комп'ютерних
технологій
Львівського національного університету
імені Івана Франка
(протокол № 47 від 29 серпня 2023 р.)

Завідувач кафедри:


Іван КАРБОВНИК

Силабус з навчальної дисципліни
“Об’єктно-орієнтоване програмування”,
що викладається в межах ОПП
“ Високопродуктивний комп’ютинг ”
першого (бакалаврського) рівня вищої освіти для здобувачів з
спеціальності 121 – Інженерія програмного забезпечення

Львів 2023 р.

Назва дисципліни	Об'єктно-орієнтоване програмування
Адреса викладання дисципліни	Корпус факультету електроніки та комп'ютерних технологій, Львівський національний університет імені Івана Франка, вул. Драгоманова 50, м. Львів, 79005, вул. Ген. Тарнавського 107, м. Львів, 79017
Факультет та кафедра, за якою закріплена дисципліна	Факультет електроніки та комп'ютерних технологій, кафедра радіофізики та комп'ютерних технологій
Галузь знань, шифр та назва спеціальності	12 – Інформаційні технології, 121 – Інженерія програмного забезпечення
Викладачі дисципліни	Кушнір Олексій Олександрович, канд. фіз.-мат. наук, доцент
Контактна інформація викладачів	oleksiy.kushnir@lnu.edu.ua, https://electronics.lnu.edu.ua/employee/kushnir-o-o
Консультації з питань навчання по дисципліні відбуваються	Консультації в день проведення лекційних занять (за попередньою домовленістю). Також можливі онлайн консультації через MS Teams. Для погодження часу онлайн консультацій слід писати на електронну пошту викладача.
Сторінка дисципліни	https://e-learning.lnu.edu.ua/course/view.php?id=4351
Інформація про дисципліну	Дисципліна «Об'єктно орієнтоване програмування» є нормативною дисципліною з спеціальності 121 Інженерія програмного забезпечення для освітньої програми «Високопродуктивний комп'ютинг», яка викладається в 2 семестрі в обсязі 4,0 кредити (за Європейською Кредитно-Трансферною Системою ECTS).
Коротка анотація дисципліни	Навчальну дисципліну розроблено таким чином, щоб надати учасникам необхідні знання та навички для написання логічних, ефективних програм мовою C++ із використанням основних конструкцій характерних для об'єктно орієнтованого програмування.
Мета та цілі дисципліни	Метою вивчення нормативної дисципліни «Об'єктно орієнтоване програмування» є одержання студентами знань з основ об'єктно-орієнтованого програмування на основі основного підтипу – клас-орієнтованого програмування алгоритмічною мовою C++. Поглиблення знань про вбудовані, структуровані типи даних та алгоритмічні конструкції цих мов. Вивчення базових принципів ООП: абстрагування, ієрархічність, інкапсулювання, успадкування та поліморфізм. Освоєння способів побудови логічної структури програми з використанням парадигми ООП.
Література для вивчення дисципліни	Основна література: 1. Стандарт мови C++ [Електронний ресурс]: https://isocpp.org/ 2. Відкритий довідник мови C++ [Електронний ресурс]: https://en.cppreference.com/w/cpp 3. Stroustrup B. The C++ Programming Language, 4th Edition, 2013, Addison-Wesley Professional 4. Josuttis N. M. C++ Standard Library, The: A Tutorial and Reference, 2012, Addison-Wesley Professional 5. Lippman S.B., Moo B.E. C++ Primer, 2012, Addison-Wesley Professional Додаткова література: 6. McConnell S. Code Complete, 2nd Edition, 2004, Cisco Press. 7. Martin R. C. Clean Code: A Handbook of Agile Software Craftsmanship, 2008, Pearson
Обсяг курсу	Загальний обсяг: 120 годин. Аудиторних занять: 64 год., з них 32 год. лекційних та 32 годин лабораторних робіт. Самостійної роботи: 56 год.

<p>Очікувані результати навчання</p>	<p>Після завершення цього курсу студент буде:</p> <ul style="list-style-type: none"> - Знати сучасні підходи до написання програмного коду мовою C++. - Вміти обирати ефективні конструкції на основі об'єктів для вирішення поставлених задач, та реалізувати їх мовою C+. <p>Після вивчення даного курсу «Об'єктно-орієнтоване програмування» здобувачі набувають таких Загальних(ЗК)/Фахових(ФК) компетентностей та Програмних результатів навчання (ПРН):</p> <p>ЗК01. Здатність до абстрактного мислення, аналізу та синтезу.</p> <p>ЗК02. Здатність застосовувати знання у практичних ситуаціях.</p> <p>ФК16. Здатність формулювати та забезпечувати вимоги щодо якості програмного забезпечення у відповідності з вимогами замовника, технічним завданням та стандартами.</p> <p>ФК17. Здатність дотримуватися специфікацій, стандартів, правил і рекомендацій в професійній галузі при реалізації процесів життєвого циклу.</p> <p>ФК25. Здатність обґрунтовано обирати та освоювати інструментарій з розробки та супроводження програмного забезпечення.</p> <p>ФК29. Здатність здійснювати розробку програмних продуктів використовуючи різні методології і технології із забезпеченням їх інноваційності та високої продуктивності.</p> <p>ПРН05. Знати і застосовувати відповідні математичні поняття, методи доменного, системного і об'єктно-орієнтованого аналізу та математичного моделювання для розробки програмного забезпечення.</p> <p>ПРН07. Знати і застосовувати на практиці фундаментальні концепції, парадигми і основні принципи функціонування мовних, інструментальних і обчислювальних засобів інженерії програмного забезпечення.</p> <p>ПРН08. Вміти розробляти людино-машинний інтерфейс.</p> <p>ПРН13. Знати і застосовувати методи розробки алгоритмів, конструювання програмного забезпечення та структур даних і знань.</p> <p>ПРН15. Мотивовано обирати мови програмування та технології розробки для розв'язання завдань створення і супроводження програмного забезпечення.</p> <p>ПРН17. Вміти застосовувати методи компонентної розробки програмного забезпечення.</p>
<p>Ключові слова</p>	<p>ООП, патерн програмування, клас, парадигми програмування</p>
<p>Формат курсу</p>	<p>Очний</p>
<p>Теми</p>	<p>Проведення лекцій, лабораторних робіт та консультації для кращого розуміння тем</p>
<p>Підсумковий контроль, форма</p>	<p>Див. СХЕМА КУРСУ</p>
<p>Пререквізити</p>	<p>Іспит в кінці семестру</p>
<p>Пререквізити</p>	<p>Для вивчення курсу студенти потребують базових знань з дисциплін «Програмування».</p>
<p>Навчальні методи та техніки, які будуть використовуватися під час викладання курсу</p>	<p>Презентація, лекції, лабораторні роботи, обговорення, дискусія, тестування, самооцінювання.</p>
<p>Необхідне обладнання</p>	<p>Мультимедіа, платформа Moodle, комп'ютерне програмне забезпечення</p> <p>Для проведення лекційних занять: комп'ютер (мінімальні характеристики: процесор Intel Core i3(4 ядра/8 потоків), 8ГБ оперативної пам'яті, 50ГБ вільного місця на диску, вбудована або</p>

	<p>дискретна відеокарта), доступ до мережі Internet, засоби мультимедіа (в т.ч. проектор).</p> <p>Для проведення лабораторних занять: Комп'ютер (мінімальні характеристики: процесор Intel Core i3(4 ядра/8 потоків), 8ГБ оперативної пам'яті, 50ГБ вільного місця на диску, вбудована або дискретна відеокарта). Необхідне програмне забезпечення включає в себе ОС Windows/Ubuntu, середовище розробки Geany/Code::Blocks/VS Code, компілятор мови програмування C++.</p>
<p>Критерії оцінювання (окремо для кожного виду навчальної діяльності)</p>	<p>Оцінювання проводиться за 100-бальною шкалою. Бали нараховуються за наступним співвідношенням:</p> <ul style="list-style-type: none"> • Змістовий модуль 1: 25% семестрової оцінки за виконані лабораторні роботи та тест модульного контролю. • Змістовий модуль 2: 25% семестрової оцінки за виконані лабораторні роботи та тест модульного контролю. • екзамен: 50% семестрової оцінки, максимальна кількість балів 50. <p>Підсумкова максимальна кількість балів 100.</p> <hr/> <p>Контрольні заміри проводяться у формі тестових завдань. Модульний контроль проводиться за допомогою тестів вибору 1 з 4 варіантів, тематика яких співпадає із темою змістового модуля та включає перевірку теоретичних знань, розуміння програмного коду та роботи компілятора. Екзамен проводиться у формі багаторівневого тесту: перший рівень – відповідь із запропонованих (1 правильна відповідь з 4), 2 рівень – відповідь цілим числом (номер рядку у якому помилка, або виконується певна дія), 3 рівень – відповідь словом, фразою, або числом, 4 рівень – есе із довільним текстом (програмний код, заданий алгоритм).</p> <p>Академічна доброчесність: Очікується, що лабораторні та контрольні 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 роботі студента є підставою для її незарахування викладачем, незалежно від масштабів плагіату чи обману.</p> <p>Відвідання занять є важливою складовою навчання. Очікується, що всі студенти відвідають усі лекції і лабораторні заняття курсу. Студенти мають інформувати викладача про неможливість відвідати заняття. Студенти зобов'язані дотримуватися усіх термінів визначених для виконання усіх видів робіт, передбачених курсом.</p> <p>Література. 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> <p>Політика виставлення балів. Враховуються бали набрані на поточному тестуванні, самостійній роботі та бали підсумкового тестування. При цьому обов'язково враховуються присутність на заняттях та активність студента під час лабораторного 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</p>

плагіат; несвоєчасне виконання поставленого завдання і т. ін.
Жодні форми порушення академічної доброчесності не толеруються.

Оцінювання лабораторних робіт (15 лабораторних робіт, максимальна кількість балів: 30) відбувається шляхом оцінки роботи студента під час проведення лабораторної роботи в аудиторії (0-5 балів за одну роботу) та захисту звіту по виконаній лабораторній роботі (0-5 балів за одну роботу). У підсумку, всі набрані бали множаться на коефіцієнт (0.2) для переведення у 40-а бальну шкалу.

Бали оцінювання лабораторних робіт нараховуються за наступним співвідношенням:

5 – студент в повному обсязі володіє навчальним матеріалом, має повне розуміння розглянутої теми, надає правильні відповіді на запитання по темі, код програми функціонує відповідно до завдання;

4 – студент достатньо розуміє розглянутий матеріал та принципи написаного ним коду програми, присутні неточності та незначні помилки у відповідях на запитання по темі, код програми функціонує відповідно до завдання (або з несуттєвими недоліками);

3 – студент не досить добре розуміє розглянутий матеріал та написаний ним код програми, вагається та надає неточні/не конкретні відповіді на запитання по темі, код програми функціонує неточно, або з помірними недоліками;

2 – студент погано розуміє розглянутий матеріал та написаний ним код програми, студент в більшості надає помилкові відповіді на питання по темі, код програми функціонує з суттєвими недоліками;

1 - студент погано розуміє розглянутий матеріал та написаний ним код програми, код програми не функціонує належним чином;

0 - студент зовсім не засвоїв розглянутий матеріал, написаний ним код програми не відповідає темі/не функціонує взагалі.

Оцінювання змістових модулів (2 змістових модулів, 10 балів за кожний) — за результатами написаних студентом есе, тестів, програм, тощо.

Бали оцінювання змістових модулів нараховуються за результатами тестування пропорційно кількості правильних відповідей, які надав студент.

Оцінювання екзамену проводиться у вигляді багаторівневого тестування, останній (четвертий) рівень якого додатково перевіряється викладачем. Оцінка за екзамен виставляється пропорційно до кількості правильних відповідей, які студент надав під час проходження тестування. Перелік тем див. у розділі **Питання до заліку чи екзамену**.

Критерії оцінювання результатів неформальної освіти:

Нарахування балів відбувається за написання студентом тез доповідей на конференціях, наукових статей, участь у діяльності наукових гуртків, участь у наукових семінарах та круглих столах, конкурсах, участь у заходах неформальної освіти за отримання сертифікатів про проходження навчання на різних освітніх платформах (Coursera, Prometheus тощо), курсах на провідних ІТ компаніях за тематикою навчальної дисципліни. Кількість балів визначається відсотком покриття результатів відповідної активності до вимог результатів

<p>Питання до заліку чи екзамену</p>	<p>навчання з навчальної дисципліни.</p> <p>Орієнтовний перелік питань та завдань для проведення підсумкової оцінки знань:</p> <ol style="list-style-type: none"> 1. Підходи до розробки програмних продуктів, їх недоліки та особливості. 2. Види та рівні мов програмування. Місце мов C та C++ серед інших сучасних мов програмування. 3. Основні положення об'єктно-орієнтованого програмування. 4. Переваги об'єктно-орієнтованої технології. Мови програмування, що реалізують ОО підхід до розробки. 5. Фундаментальні поняття об'єктно-орієнтованого програмування. Поняття успадкування. Ієрархія об'єктів. 6. Фундаментальні поняття об'єктно-орієнтованого програмування. Абстрагування. Поліморфізм. Приховання інформації (інкапсуляція). 7. Особливості об'єктно-орієнтовного підходу у C++. 8. Поняття об'єкту та поняття класу. Поняття клас-орієнтовного програмування, схожість та не ідентичність з ООП. 9. Елементи об'єктної моделі у C++. Поняття атрибутів (полів, методів). 10. Структура класу у C++. Екземпляр класу. Опис полів та методів класу. 11. Класифікація методів. Особливі методи – конструктор та деструктор. 12. Реалізація успадкування та поліморфізму у C++. 13. Деталізація поняття конструктору, види конструкторів класу, копіюючий конструктор. 14. Просте та множинне успадкування. Перевантаження методів. 15. Локальні та глобальні класи. Вкладені класи. Приклади використання. 16. Неявний вказівник this. Правила використання вказівника this. 17. Дружні функції та їх використання. 18. Поняття динамічної бібліотеки DLL. Особливості роботи з динамічними бібліотеками у C++. 19. Файли, які необхідні для використання бібліотек DLL, їх структура. 20. Структура та правила використання потужного інструменту C++ -- лямбда вирази. 21. Випадки, у яких необхідно відкривати новий потік. Створення та закриття потоків. Керування пам'яттю. 22. Створення та опрацювання івентів. Керування потоками за допомогою івентів. 23. Асинхронні та синхронні потоки. Використання класів через вказівники. 24. Що таке SOLID, як він пов'язаний із ООП. 25. Шаблони проєктування (патерни).
<p>Опитування</p>	<p>Анкету-оцінку з метою оцінювання якості курсу буде надано по завершенню курсу.</p>

СХЕМА КУРСУ

Тиж.	Тема, план, короткі тези	Форма діяльності (заняття)	Література. Ресурси в Інтернеті	Завдання (лабораторна робота), год	Термін виконання
1	Різні підходи до програмування у історичному контексті. Еволюція мов програмування від зародження обчислювальної техніки до сьогодення. Різні підходи до розробки програмних продуктів, їх недоліки та особливості. Місце мов С та С++ серед інших сучасних мов програмування.	Лекція	1, 4, Сайт курсу - https://e-learning.lnu.edu.ua/course/view.php?id=4351	2	1 тиж. семестру
1	Вступне заняття. Інструкція з техніки безпеки, перевірка обладнання та налаштування програмного забезпечення.	Лабораторна робота	1, 2, Сайт курсу - https://e-learning.lnu.edu.ua/course/view.php?id=4351	2	1 тиж. семестру
2	Відмінності С та С++. Основні відмінності між мовами С та С++. Відмінності у роботі та синтаксисі мов. Нові ключові слова, які застосовуються у С++ та їхнє значення. Конструкції у С++, яких немає у С. Наявність недоліків та обмежень у С++ (відносно мови С).	Лекція	1, 3, 4, Сайт курсу - https://e-learning.lnu.edu.ua/course/view.php?id=4351	2	2 тиж. семестру
2	Структури та об'єднання у мовах С та С++	Лабораторна робота	2, Сайт курсу - https://e-learning.lnu.edu.ua/course/view.php?id=4351	2	2 тиж. семестру
3	Основні принципи об'єктно-орієнтованого підходу. Основні положення ООП. Засади успіху ООП технології у розробці програмних продуктів. Загальні переваги ООП технології. Мови програмування, що реалізують ОО підхід до розробки.	Лекція	1, 2, 3, 4	2	3 тиж. семестру
3	Особливості та відмінності вводу/виводу даних у мовах С та С++. Розв'язок задачі з використанням структур та об'єднань	Лабораторна робота	1, Сайт курсу - https://e-learning.lnu.edu.ua/course/view.php?id=4351	2	3 тиж. семестру
4-5	Фундаментальні поняття об'єктно-орієнтованого програмування. Поняття успадкування. Ієрархія об'єктів. Абстрагування. Поліморфізм. Приховання інформації (інкапсуляція).	Лекція	2, 3, Сайт курсу - https://e-learning.lnu.edu.ua/course/view.php?id=4351	4	4-5 тиж. семестру
4-5	Створення власного класу. Реалізація алгоритму за заданим шаблоном.	Лабораторна робота	1, 2, Сайт курсу - https://e-learning.lnu.edu.ua/course/view.php?id=4351	4	4-5 тиж. семестру
6-7	Особливості об'єктно-орієнтованого підходу у С++. Поняття об'єкту та поняття класу. Поняття клас-орієнтованого програмування, схожість та не ідентичність з ООП. Елементи об'єктної моделі у С++. Поняття	Лекція	3, Сайт курсу - https://e-learning.lnu.edu.ua/course/view.php?id=4351	4	6-7 тиж. семестру

	атрибутів (полів, методів).				
6	Реалізація різних випадків спадкування та поліморфізму мовою C++.	Лабораторна робота	2, Сайт курсу - https://e-learning.lnu.edu.ua/course/view.php?id=4351	2	6 тиж. семестру
7	Множинне успадкування, особливості його реалізації у C++. Проблема ромба	Лабораторна робота	2, Сайт курсу - https://e-learning.lnu.edu.ua/course/view.php?id=4351	2	7 тиж. семестру
8	Класи у C++. Структура класу у C++. Екземпляр класу. Опис полів та методів класу. Класифікація методів. Особливі методи – конструктор та деструктор.	Лекція	7, Сайт курсу - https://e-learning.lnu.edu.ua/course/view.php?id=4351	2	8 тиж. семестру
8-9	Створення та використання динамічних бібліотек	Лабораторна робота	Сайт курсу - https://e-learning.lnu.edu.ua/course/view.php?id=4351	4	8-9 тиж. семестру
9	Реалізація успадкування та поліморфізму у C++. Деталізація поняття конструктору, види конструкторів класу, копіюючий конструктор. Приклад використання копіюючого конструктору. Просте та множинне успадкування. Перевантаження методів.	Лекція	2, Сайт курсу - https://e-learning.lnu.edu.ua/course/view.php?id=4351	2	9 тиж. семестру
10	Види класів. Локальні та глобальні класи. Вкладені класи. Приклади використання. Неявний вказівник this. Правила використання вказівника this. Дружні функції та їх використання.	Лекція	4, 5, Сайт курсу - https://e-learning.lnu.edu.ua/course/view.php?id=4351	2	10 тиж. семестру
10	Лямбда вирази	Лабораторна робота	1, 2, Сайт курсу - https://e-learning.lnu.edu.ua/course/view.php?id=4351	2	10 тиж. семестру
11	Динамічні бібліотеки та їх використання. Поняття динамічної бібліотеки DLL. Особливості роботи з динамічними бібліотеками у C++. Файли, які необхідні для використання бібліотек DLL, їх структура.	Лекція	5, Сайт курсу - https://e-learning.lnu.edu.ua/course/view.php?id=4351	2	11 тиж. семестру
11-12	Використання паралельних потоків для створення секундоміра	Лабораторна робота	Сайт курсу - https://e-learning.lnu.edu.ua/course/view.php?id=4351	4	11-12 тиж. семестру
12	Лямбда вирази у C++. Структура та правила використання потужного інструменту C++ -- лямбда вирази.	Лекція	2, 3, Сайт курсу - https://e-learning.lnu.edu.ua/course/view.php?id=4351	2	12 тиж. семестру
13	Використання вказівників на клас	Лабораторна робота	Сайт курсу - https://e-learning.lnu.edu.ua/course/view.php?id=4351	2	13 тиж. семестру
13	Потоки. Випадки, у яких необхідно відкривати новий потік.	Лекція	3, Сайт курсу - https://e-learning.lnu.edu.ua/course/view.php?id=4351	2	13 тиж. семестру

	Створення та закриття потоків. Керування пам'яттю.				
14	Зчитування даних з звукової карти	Лабораторна робота	Сайт курсу - https://e-learning.lnu.edu.ua/course/view.php?id=4351	2	14 тиж. семестру
14	Івенти. Створення та опрацювання івентів. Керування потоками за допомогою івентів. Ассинхронні та синхронні потоки. Використання класів через вказівники.	Лекція	4, Сайт курсу - https://e-learning.lnu.edu.ua/course/view.php?id=4351	2	14 тиж. семестру
15-16	Застосування шаблонів проектування для вирішення стандартних задач	Лабораторна робота	Сайт курсу - https://e-learning.lnu.edu.ua/course/view.php?id=4351	4	15-16 тиж. семестру
15-16	Концепція SOLID Що таке SOLID, як він пов'язаний із ООП. Шаблони проектування	Лекція	1, 2, 3, Сайт курсу - https://e-learning.lnu.edu.ua/course/view.php?id=4351	4	15-16 тиж. семестру