

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**  
**Львівський національний університет імені Івана Франка**  
**Факультет електроніки та комп'ютерних технологій**  
**Кафедра радіоелектронних і комп'ютерних систем**

**Затверджено**

На засіданні  
кафедри радіоелектронних і  
комп'ютерних систем  
факультету електроніки та комп'ютерних  
технологій  
Львівського національного університету  
імені Івана Франка  
(протокол №1 від 31.08.2022 р.)

Завідувач кафедри:



Ігор Оленич

**Силабус з навчальної дисципліни**  
**“Патерни проектування”,**  
**що викладається в межах ОПП**  
**“Інженерія програмного забезпечення”**  
**першого (бакалаврського) рівня вищої освіти для здобувачів з**  
**спеціальності 121 – Інженерія програмного забезпечення**

Львів 2022 р.

<b>Назва дисципліни</b>	Патерни проектування
<b>Адреса викладання дисципліни</b>	Корпус факультету електроніки та комп'ютерних технологій, Львівський національний університет імені Івана Франка, вул. Драгоманова 50, м. Львів, 79005
<b>Факультет та кафедра, за якою закріплена дисципліна</b>	Факультет електроніки та комп'ютерних технологій Кафедра радіоелектронних і комп'ютерних систем
<b>Галузь знань, шифр та назва спеціальності</b>	12 – інформаційні технології 121 – Інженерія програмного забезпечення
<b>Викладачі дисципліни</b>	Оленич І.Б., докт. фіз.-мат. наук, професор Сінкевич О. О., асистент радіоелектронних і комп'ютерних систем
<b>Контактна інформація викладачів</b>	<a href="mailto:igor.olenych@lnu.edu.ua">igor.olenych@lnu.edu.ua</a> <a href="mailto:oleh.sinkevych@lnu.edu.ua">oleh.sinkevych@lnu.edu.ua</a> , <a href="https://electronics.lnu.edu.ua/employee/o_sinkevych">https://electronics.lnu.edu.ua/employee/o_sinkevych</a>
<b>Консультації з питань навчання по дисципліні відбуваються</b>	Консультації в день проведення лекцій/лабораторних занять (за попередньою домовленістю): ауд. 102, корпус факультету електроніки та комп'ютерних технологій, вул. Драгоманова 50, м. Львів
<b>Сторінка курсу</b>	<a href="https://electronics.lnu.edu.ua/course/paterny-proektuvannia-ipz">https://electronics.lnu.edu.ua/course/paterny-proektuvannia-ipz</a> <a href="https://e-learning.lnu.edu.ua/course/view.php?id=4405">https://e-learning.lnu.edu.ua/course/view.php?id=4405</a>
<b>Коротка анотація дисципліни</b>	Дисципліна “Патерни проектування” є нормативною дисципліною з спеціальності 121 – Інженерія програмного забезпечення для освітньої програми “Інформаційні системи та технології”, яка викладається в 3-му семестрі в обсязі 6-ох кредитів (за Європейською Кредитно-Трансферною Системою ECTS).
<b>Інформація про дисципліну</b>	Курс розроблено для ознайомлення студентів з основними поняттями та патернами проектування об'єктно-орієнтованих систем, класифікацією та особливостями патернів проектування та підходами до їх використання при розробці програмних засобів.
<b>Мета та цілі дисципліни</b>	<i>Мета:</i> одержання студентами знань про основні види патернів проектування; об'єднання об'єктно-орієнтованого коду у контексті породжувальних, структурних і поведінкових патернів; процес побудови структурних одиниць патернів; стандарти оформлення об'єктно-орієнтованого коду; механізми взаємодії складної структури класів. <i>Цілі:</i> навчити студентів реалізовувати і застосування патерни проектування; ознайомити студентів з основними видами патернів: структурного, породжувального і поведінкового; забезпечити розуміння проблематики та своєчасності застосування конкретних патернів на багатьох прикладах; показати нюанси і схеми конкретних патернів і дати навички для їх використання у проектування типових рішень в межах об'єктно-орієнтованої парадигми.
<b>Література для вивчення дисципліни</b>	1) Erich Gamma, John Vlissides, Richard Helm, Ralph Johnson. Design patterns, software engineering, object-oriented programming. Addison-Wesley, 1994. – 335 p.

	<p>2) Ерік Фрімен. Патерни проектування. Фабула, 2020. – 672 с.</p> <p>3) Sean Bradley. Design Patterns In Python. Common GoF (Gang of Four) Design Patterns Implemented In Python, 2021. – 238 p.</p> <p>4) Robert Martin. Clean Code: A Handbook of Agile Software Craftsmanship. Pearson Education (US), 2008. – 464 p.</p> <p>5) John F. Dooley. Software Development, Design and Coding. Apress, 2017. – 330 p.</p> <p>6) Швець О. Refactoring and design patterns. <i>Refactoring and Design Patterns</i>. URL: <a href="https://refactoring.guru/">https://refactoring.guru/</a>.</p> <p>7) Kasun Indrasiri. Design Patterns for Cloud Native Applications: Patterns in Practice Using APIs, Data, Events, and Streams. O'Reilly Media, 2021. – 314 с.</p> <p>8) J Geewax. API Design Patterns. Manning Publications Co, 2021. - 480 с.</p>
<p><b>Обсяг курсу</b></p>	<p>Загальний обсяг: 120 годин. Аудиторних занять: 64 год., з них 32 год. лекційних та 32 годин лабораторних робіт. Самостійної роботи: 56 год.</p>
<p><b>Очікувані результати навчання</b></p>	<p>У результаті вивчення даного курсу студент буде:</p> <p><b>знати:</b> що таке патерн проектування і з чого він складається; класифікацію патернів, історію їх виникнення та фундаментальні джерела; принципи проектування та якості правильної архітектури коду; інтерфейси класів; принцип SOLID; коли і як застосовувати конкретну реалізацію патерна.</p> <p><b>вміти:</b> правильно оформляти об'єктно-орієнтований код; застосовувати засвоєний матеріал для створення об'єктно-орієнтованих програм; використовувати Python для програмної реалізації патернів; реалізувати просте модульне тестування; визначати, коли необхідно використовувати патерн певного типу.</p> <p>Після вивчення даного курсу «Патерни проектування» здобувачі набудуть таких Загальних та Фахових компетентностей та Програмних результатів навчання:</p> <p>ЗК1. Здатність до абстрактного мислення, аналізу та синтезу.</p> <p>ЗК2. Здатність застосовувати знання у практичних ситуаціях.</p> <p>ЗК4. Здатність спілкуватися іноземною мовою як усно, так і письмово.</p> <p>ФК14. Здатність брати участь у проектуванні програмного забезпечення, включаючи проведення моделювання (формальний опис) його структури, поведінки та процесів функціонування.</p> <p>ФК15. Здатність розробляти архітектури, модулі та компоненти програмних систем.</p> <p>ФК17. Здатність дотримуватися специфікацій, стандартів, правил і рекомендацій в професійній галузі при реалізації процесів життєвого циклу.</p> <p>ФК20. Здатність застосовувати фундаментальні і міждисциплінарні знання для успішного розв'язання завдань інженерії програмного забезпечення.</p> <p>ФК22. Здатність накопичувати, обробляти та систематизувати професійні знання щодо створення і супроводження програмного забезпечення та визнання важливості навчання протягом всього життя.</p> <p>ФК23. Здатність реалізовувати фази та ітерації життєвого циклу програмних систем та інформаційних технологій на основі відповідних моделей і підходів розробки програмного забезпечення.</p> <p>ФК24. Здатність здійснювати процес інтеграції системи, застосовувати стандарти і процедури управління змінами для підтримки цілісності, загальної функціональності і надійності програмного забезпечення.</p>

	<p>ФК25. Здатність обґрунтовано обирати та освоювати інструментарій з розробки та супроводження програмного забезпечення.</p> <p>ФК29. Здатність здійснювати розробку програмного забезпечення використовуючи сучасні парадигми програмування.</p> <p>ПРН5. Знати і застосовувати відповідні математичні поняття, методи доменного, системного і об'єктно-орієнтованого аналізу та математичного моделювання для розробки програмного забезпечення.</p> <p>ПРН7. Знати і застосовувати на практиці фундаментальні концепції, парадигми і основні принципи функціонування мовних, інструментальних і обчислювальних засобів інженерії програмного забезпечення.</p> <p>ПРН8. Вміти розробляти людино-машинний інтерфейс.</p> <p>ПРН12. Застосовувати на практиці ефективні підходи щодо проектування програмного забезпечення.</p> <p>ПРН13. Знати і застосовувати методи розробки алгоритмів, конструювання програмного забезпечення та структур даних і знань.</p> <p>ПРН14. Застосовувати на практиці інструментальні програмні засоби доменного аналізу, проектування, тестування, візуалізації, вимірювань та документування програмного забезпечення.</p> <p>ПРН15. Мотивовано обирати мови програмування та технології розробки для розв'язання завдань створення і супроводження програмного забезпечення.</p> <p>ПРН27. Вміти обирати оптимальні алгоритми та технології розробки програмного забезпечення.</p>
<b>Ключові слова</b>	Патерни проектування, об'єктно-орієнтоване програмування, SOLID, принципи проектування, рефакторинг.
<b>Формат курсу</b>	Очний
<b>Теми</b>	<p>Тема 1. <i>Вступ до ООП. Парадигми програмування. Особливості об'єктів.</i></p> <p>Тема 2. <i>Патерни та принципи проектування. SOLID.</i></p> <p>Тема 3. <i>Робота з атрибутами. Абстрактний та базовий класи.</i></p> <p>Тема 4. <i>Патерни фабричний метод та абстрактна фабрика.</i></p> <p>Тема 5. <i>Патерни фабрик. Патерн будівельник.</i></p> <p>Тема 6. <i>Змінні та незмінні типи. Патерн прототип. Метаклас.</i></p> <p>Тема 7. <i>Проектування метакласів. Патерн одинак.</i></p> <p>Тема 8. <i>Патерн адаптер. Вступ до декораторів.</i></p> <p>Тема 9. <i>Патерн декоратор. Патерн фасад.</i></p> <p>Тема 10. <i>Поглиблення структурних патернів. Міст.</i></p> <p>Тема 11. <i>Патерн проксі. Компонувальник та дерево.</i></p> <p>Тема 12. <i>Вступ до поведінкових патернів. Патерн стратегія.</i></p> <p>Тема 13. <i>Патерн спостерігач.</i></p> <p>Тема 14. <i>Стейт-машина. Патерн стан.</i></p> <p>Тема 15. <i>Веб-патерни. Вступ до архітектурного патерну модель-вид-контролер ч.1.</i></p> <p>Тема 16. <i>MVC патерн проектування та приклади розробки.</i></p>
<b>Підсумковий контроль, форма</b>	Залік у кінці семестру
<b>Пререквізити</b>	<p>Для вивчення даного курсу студентам потрібні базові знання з курсів:</p> <ul style="list-style-type: none"> <li>- основи програмування;</li> <li>- теорія алгоритмів;</li> <li>- алгоритмізація і програмування;</li> </ul>

	- об'єктно-орієнтоване програмування.																																												
<b>Навчальні методи та техніки, які будуть використовуватися під час викладання курсу</b>	Інформаційні методи (лекція, бесіда, ілюстрація, демонстрація); дедуктивні методи на основі узагальнень; евристичні методи (проблемна лекція); інтерактивні методи (дискусія).																																												
<b>Необхідне обладнання</b>	Комп'ютер із необхідним програмним забезпеченням, доступ до Internet мережі.																																												
<b>Критерії оцінювання (окремо для кожного виду навчальної діяльності)</b>	<p>Оцінювання проводиться за 100-бальною шкалою. Бали нараховуються за наступним співвідношенням:</p> <ul style="list-style-type: none"> <li>• Змістовий модуль 1: 20% семестрової оцінки за виконане модульне завдання, максимальна кількість балів 20.</li> <li>• Змістовий модуль 2: 20% семестрової оцінки за виконані лабораторні роботи, максимальна кількість балів 20.</li> <li>• Лабораторні роботи: 60% семестрової оцінки, максимальна кількість балів 60.</li> </ul> <p>Підсумкова максимальна кількість балів 100.</p> <table border="1" style="width: 100%; border-collapse: collapse; text-align: center;"> <thead> <tr> <th colspan="10">Поточне тестування та самостійна робота</th> <th rowspan="2">Сума</th> </tr> <tr> <th colspan="8">Лабораторні роботи</th> <th>Змістовий модуль 1</th> <th>Змістовий модуль 2</th> <th rowspan="2"></th> </tr> <tr> <th>Л1</th> <th>Л2</th> <th>Л3</th> <th>Л4</th> <th>Л5</th> <th>Л6</th> <th>Л7</th> <th>Л8</th> <th>М1</th> <th>М2</th> <th></th> </tr> </thead> <tbody> <tr> <td>7.5</td> <td>7.5</td> <td>7.5</td> <td>7.5</td> <td>7.5</td> <td>7.5</td> <td>7.5</td> <td>7.5</td> <td>20</td> <td>20</td> <td>100</td> </tr> </tbody> </table> <p><b>Академічна доброчесність:</b> Очікується, що роботи студентів будуть оригінальними дослідженнями чи міркуваннями. Списування та втручання в роботу інших студентів становлять, але не обмежують, приклади можливої академічної недоброчесності. Виявлення ознак академічної недоброчесності в написанні завдань є підставою для її незарахування викладачем, незалежно від масштабів плагіату чи обману.</p> <p><b>Відвідання занять</b> є важливою складовою навчання. Очікується, що всі студенти відвідають усі лекції та лабораторні заняття курсу. Студенти повинні інформувати викладача про неможливість відвідати заняття. У будь-якому випадку студенти зобов'язані дотримуватися термінів визначених для виконання всіх видів робіт, передбачених курсом.</p> <p><b>Література.</b> 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><b>Політика виставлення балів.</b> Враховуються бали, набрані при поточному контролі та бали підсумкового тестування. При цьому обов'язково враховуються присутність на заняттях та активність студента під час практичного заняття; недопустимість пропусків та запізнень 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. Жодні форми порушення академічної доброчесності не толеруються.</p>	Поточне тестування та самостійна робота										Сума	Лабораторні роботи								Змістовий модуль 1	Змістовий модуль 2		Л1	Л2	Л3	Л4	Л5	Л6	Л7	Л8	М1	М2		7.5	7.5	7.5	7.5	7.5	7.5	7.5	7.5	20	20	100
Поточне тестування та самостійна робота										Сума																																			
Лабораторні роботи								Змістовий модуль 1	Змістовий модуль 2																																				
Л1	Л2	Л3	Л4	Л5	Л6	Л7	Л8	М1	М2																																				
7.5	7.5	7.5	7.5	7.5	7.5	7.5	7.5	20	20	100																																			
<b>Питання до</b>	Об'єктно-орієнтоване програмування																																												

<b>заліку чи екзамену.</b>	Поняття об'єкту, класу Парадигми ООП. Інкапсуляція та поліморфізм. Інтерфейс та стан об'єкту Базовий та абстрактний класи. Успадкування Принципи та підходи до проектування ООП систем Класифікація патернів проектування Група патернів фабрика. Ідіома Фабричний метод та абстрактна фабрика Патерн будівельник Патерн одинак Патерн прототип. Змінні та незмінні типи. Метакласи та їх особливості Структурні патерни. Адаптер Патерн Міст та фасад Декоратори функцій. Патерн декоратор Поведінкові патерни. Патерн спостерігач Патерн стратегія Патерн стан та шаблонний метод Архітектурні патерни. Клієнт-серверна взаємодія Патерн Модель — Вид — Контролер.
<b>Опитування</b>	Анкету-оцінку з метою оцінювання якості курсу буде надано по завершенню курсу.

**Схема курсу “Патерни проектування”  
для студентів спеціальності 121 – Інженерія програмного забезпечення**

Тиж.	Тема, план, короткі тези	Форма діяльності (заняття)* *лекція, самостійна, дискусія, групова робота)	Література.** * Ресурси в інтернеті	Завдання, год	Термін виконання
1	Огляд об'єктно-орієнтованого програмування та основних засад.	Лекція	[1], [2], [3], [8] Сайт курсу	2	кінець поточного тижня
1	Лаб. 1. Робота з класами. Реалізація взаємозв'язків між об'єктами.	Лабораторна робота	Сайт курсу	2	протягом двох тижнів
2	Патерни та принципи проектування. SOLID та Роберт Мартін.	Лекція	[1], [2], [3] Сайт курсу	2	кінець поточного тижня
2	Опрацювання книжки Роберта Мартіна — Чистий код (перші три глави)	Самостійна робота	[4]	2	кінець поточного тижня
3	Управління атрибутами. Абстрактний клас та абстрактні методи	Лекція	[1], [2], [3], Сайт курсу	2	кінець поточного тижня
3	Лаб. 2. Наслідування та абстрактні класи	Лабораторна робота	Сайт курсу	2	протягом двох тижнів
4	Фабричний метод та абстрактна фабрика	Лекція	[1], [2], [3], [6] Сайт курсу	2	кінець поточного тижня
4	Перегляд та засвоєння реалізації патернів (фабрики) різними мовами	Самостійна робота	[6], [8]	2	кінець поточного тижня
5	Патерн будівельник. Поєднання будівельника та фабрики	Лекція	[1], [2], [3], [6] Сайт курсу	2	кінець поточного тижня
5	Лаб. 3. Реалізація фабричного методу та абстрактної фабрики	Лабораторна робота	Сайт курсу	2	протягом двох тижнів
6	Змінні та незмінні типи. Прототип. Введення у метакласи	Лекція	[1], [2], [3], [6] Сайт курсу	2	кінець поточного тижня
6	Перегляд та засвоєння реалізації патерну прототип різними мовами. Ознайомлення з додатковими матеріалами	Самостійна робота	[6], [8]	2	кінець поточного тижня
7	Метакласи та їх використання у патерні одинак. Приклади	Лекція	[1], [2], [3], [6] Сайт курсу	2	кінець поточного тижня
7	Лаб. 4. Реалізація патернів будівельник та прототип	Лабораторна робота	Сайт курсу	2	протягом двох тижнів
8	Структурні патерни. Фасад. Основи декораторів	Лекція	[1], [2], [3], [6] Сайт курсу	2	кінець поточного тижня
8	Перегляд та засвоєння реалізації патерну фасад різними мовами. Ознайомлення з додатковими матеріалами	Самостійна робота	[1], [2], [3], [6] Сайт курсу	2	кінець поточного тижня
9	Патерн декоратор та патерн фасад	Лекція	[1], [2], [3], [6] Сайт курсу	2	кінець поточного

					тижня
9	Модульне завдання (породжувані та структурні патерни)	Модульне завдання	Сайт курсу	2	кінець поточного тижня
10	Поглиблення у структурні патерни. Патерн міст.	Лекція	[1], [2], [3], [6] Сайт курсу	2	кінець поточного тижня
10	Лаб. 5. Реалізація декоратора та адаптера	Лабораторна робота	Сайт курсу	2	протягом двох тижнів
11	Патерн проксі. Компонувальник та дерево.	Лекція	[1], [2], [3], [6] Сайт курсу	2	кінець поточного тижня
11	Перегляд та засвоєння реалізації патерну проксі та компонентувальник різними мовами. Ознайомлення з додатковими матеріалами	Самостійна робота	[1], [2], [3], [6] Сайт курсу	2	кінець поточного тижня
12	Вступ до поведінкових патернів. Патерн стратегія.	Лекція	[1], [2], [3], [6] Сайт курсу	2	кінець поточного тижня
12	Лаб. 6. Реалізація патернів фасад та міст	Лабораторна робота	Сайт курсу	2	протягом двох тижнів
13	Патерн спостерігач. Приклади	Лекція	[1], [2], [3], [6] Сайт курсу	2	кінець поточного тижня
13	Перегляд та засвоєння реалізації патерну стратегія та спостерігач різними мовами. Ознайомлення з додатковими матеріалами	Самостійна робота	[1], [2], [3], [6] Сайт курсу	2	кінець поточного тижня
14	Стейт-машина. Патерн стан.	Лекція	[1], [2], [3], [6] Сайт курсу	2	кінець поточного тижня
14	Лаб. 7. Реалізація патернів проксі, стратегії та спостерігача	Лабораторна робота	Сайт курсу	2	протягом двох тижнів
15	Веб-патерни. Вступ до архітектурного патерну модель-вид-контролер ч.1.	Лекція	[1], [2], [3], [6] Сайт курсу	2	кінець поточного тижня
15	Модульне завдання (структурні та поведінкові патерни)	Модульне завдання	Сайт курсу	2	кінець поточного тижня
16	MVC патерн проектування та приклади розробки	Лекція	[1], [2], [3], [8] Сайт курсу	2	кінець поточного тижня
16	Лаб. 8. Реалізація простого MVC застосунку з використанням фреймворку на вибір студента	Лабораторна робота	Сайт курсу	2	кінець поточного тижня