



МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Львівський національний університет імені Івана Франка
Факультет управління фінансами та бізнесу
Кафедра цифрової економіки та бізнес-аналітики

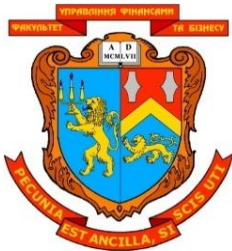
ЗАТВЕРДЖЕНО

На засіданні кафедри цифрової економіки та
бізнес-аналітики
факультету управління фінансами та бізнесу
Львівського національного університету імені
Івана Франка
(протокол № 6 від 16 січня 2024 р.)

Завідувач кафедри _____ Ірина ШЕВЧУК

Силабус з навчальної дисципліни
«Об'єктно-орієнтований дизайн»,
що викладається в межах ОПШ
«Інформаційні технології в бізнесі»
другого (магістерського) рівня вищої освіти для здобувачів з
спеціальності 051 «Економіка»

Львів 2024 р.

	<p style="text-align: center;">Силабус навчальної дисципліни «ОБ'ЄКТНО-ОРІЄНТОВАНИЙ ДИЗАЙН» Галузь знань: 05 «Соціальні та поведінкові науки» Спеціальність: 051 «Економіка»</p>
Адреса викладання дисципліни	м. Львів, вул. Коперника, 3
Факультет та кафедра, за якою закріплена дисципліна	Факультет управління фінансами та бізнесу Кафедра цифрової економіки та бізнес-аналітики
Галузь знань, шифр та назва спеціальності	05 «Соціальна та поведінкові науки» 051 «Економіка»
Викладачі дисципліни	Шевчук Ірина Богданівна, д.е.н., професор, завідувач кафедри цифрової економіки та бізнес-аналітики
Контактна інформація викладачів	Моб. телефон: +38(067)-77-39-777 Електронні скриньки: ibshevchuk@ukr.net; iryna.shevchuk@lnu.edu.ua Viber: 067-77-39-777; Telegram: Shevchuk Iryna, 067-77-39-777 Messenger: Iryna Shevchuk; Skype: ibshevchuk Сторінка викладача: https://financial.lnu.edu.ua/employee/shevchuk-iryna-bohdanivna Місце знаходження: м. Львів, вул. Коперника, 3; кім. 508 (кафедра цифрової економіки та бізнес-аналітики)
Консультації з питань навчання по дисципліні відбуваються	Щосереди, 15:00-16:20 год. (вул. Коперника, 3, ауд. 302) Консультації в день проведення лекцій/лабораторних занять (за попередньою домовленістю). Можливі он-лайн консультації через Skype, Viber, Telegram. Для погодження часу он-лайн консультацій слід писати на електронну пошту викладача або дзвонити.
Сторінка курсу	https://financial.lnu.edu.ua/course/ob-iektno-orientovanyy-dyzayn Платформа MOODLE: http://e-learning.lnu.edu.ua/login/index.php
Інформація про дисципліну	Курс розроблено таким чином, щоб надати здобувачам вищої освіти необхідні знання для набуття і прикладного використання компетентностей, обов'язкових для того, щоб стати фахівцем із розробки складних програмних систем з сучасним інтерфейсом, а також посісти конкурентоздатну позицію на ринку праці. Тому у курсі розглянуто основні методології програмування, концепцію та принципи об'єктно-орієнтованого програмування, основи об'єктно-орієнтованого аналізу, принципи об'єктно-орієнтованого дизайну, механізм об'єктно-реляційного відображення та прикладні аспекти використання уніфікованої мови моделювання UML.
Коротка анотація дисципліни	Дисципліна «Об'єктно-орієнтований дизайн» є вибірковою дисципліною зі спеціальності 051 «Економіка» для освітньої програми «Інформаційні технології в бізнесі», яка викладається в II семестрі в обсязі 3 кредити (за Європейською Кредитно-Трансферною Системою ECTS).
Мета та цілі дисципліни	Метою вивчення нормативної дисципліни «Об'єктно-орієнтований дизайн» є ознайомлення студентів із сучасними методами проектування програмного забезпечення, що дозволяють здійснювати розробку програмних систем середньої та високої складності.

	<p>Основні завдання дисципліни «Об'єктно-орієнтований дизайн» – ознайомлення студентів із сучасними процесами розробки програмного забезпечення (ПЗ), орієнтованими на наскрізне проектування; вивчення основ мови моделювання UML; вивчення принципів аналізу та проектування об'єктно-орієнтованих систем; вивчення найбільш поширених шаблонів проектування та архітектурних шаблонів.</p>
<p>Література для вивчення дисципліни</p>	<p>Основна:</p> <ol style="list-style-type: none"> 1. Ajit Singh, Ms. Anamika. Object Oriented Modeling and Design Using UML. 2nd Edition. Independently published. 2023. 153 p. 2. Craig Larman. Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development. Pearson, 2014. 616 p. 3. Dean Leffingwell, Don Widrig. Managing Software Requirements: A Unified Approach. Addison-Wesley Professional, 2003. 560 p. 4. Grady Booch, James Rumbaugh, Ivar Jacobson. The Unified Modeling Language User Guide. Addison-Wesley Professional, 2005. 496 p. 5. Grady Booch, Robert A. Maksimchuk, Michael W. Engel, Bobbi J. Young, Jim Conallen, Kelli A. Houston. Object-Oriented Analysis and Design with Applications. Addison-Wesley Professional, 2007. 720 p. 6. James Rumbaugh, Ivar Jacobson, Grady Booch The Unified Modeling Language Reference Manual 2nd Edition. Addison-Wesley Professional; 2nd edition. 2004. 721 p. 7. Janis Osis, Uldis Donins. Approach for Domain Modeling and Software Development. Elsevier Science Publishing Co Inc. 2017. 276 p. 8. Jean-Marc Nerson. Enterprise Model Patterns: Describing the World. Technics Publications, 2014. 648 p. 9. Martin Fowler. UML Distilled: A Brief Guide to the Standard Object Modeling Language. Addison-Wesley Professional, 2003. 208 p. 10. Pender T.A. UML Weekend Crash Course. Wiley Publishing Inc., 2002. 358 p. 11. Stephen J. Mellor, Marc J. Balcer, Ivar Jacobson. Executable UML: A Foundation for Model-Driven Architecture. Independently published. 2022. 402 p. 12. Авраменко В.С., Авраменко А.С. Проектування інформаційних систем: навчальний посібник / В.С. Авраменко, А.С. Авраменко. Черкаси: Черкаський національний університет ім. Б. Хмельницького, 2017. 434 с. 13. Омельчук Л.Л. Об'єктно-орієнтоване програмування. Лабораторний практикум: навчальний посібник / Л.Л. Омельчук, А.С. Белова. Київ: 2022. 271 с. 14. Роберт С. Мартін. Чиста архітектура. Фабула. 2019. 368 с. <p>Додаткова:</p> <ol style="list-style-type: none"> 1. Dean Leffingwell, Don Widrig. Managing Software Requirements: A Unified Approach. Addison-Wesley Professional, 2003. 560 p. 2. Grady Booch, James Rumbaugh, Ivar Jacobson. The Unified Modeling Language User Guide. Addison-Wesley Professional, 2005. 496 p. 3. Krishna Rungta. UML 2.0 : Learn UML in 1 Day. Independently Published. 2019. 112 p. 4. Larman Craig. Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development. Pearson, 2004. 616 p. 5. Norman Daoust. UML Requirements Modeling for Business Analysts : Steps to Modeling Success. Technics Publications LLC. 2012. 268 p. 6. Pender T.A. UML Weekend Crash Course. Wiley Publishing Inc., 2002. 358 p. 7. Scott W. Ambler. The Object Primer : Agile Model-Driven Development with UML 2.0. Cambridge University Press. 2015. 572 p. 8. Suriya Sundaramoorthy. UML Diagramming: A Case Study Approach Auerbach Publications; 1st edition. 416 p. 9. Інструментальні програмні засоби розробки ІУС. Методичні вказівки до виконання лабораторних робіт. / уклад.: К.І. Київська. – Київ: КНУБА, 2018. – 40 с.

	<p>10. Райчев І.Е., Харченко О.Г., Замковий В.В. Принципи проектування відкритих розподілених систем : навч. посіб. -К.: Вид-во Нац. авіац. ун-ту "НАУ-друк", 2010. 240 с</p> <p>Інтернет-джерела:</p> <ol style="list-style-type: none"> 1. Curated list of UML tools - 2022 edition. URL: https://modeling-languages.com/uml-tools/ 2. Introduction to business modeling using the Unified Modeling Language (UML) [Електронний ресурс] // IBM. - 2003. URL: http://www.ibm.com/developerworks/rational/library/360.html 3. The Unified Modeling Language Reference Manual. URL: https://people.ucalgary.ca/~far/Lectures/SENG609-23/PDF/uml_refman.pdf 4. UML Java Programmers. URL: https://www.csd.uoc.gr/~hy252/references/UML_for_Java_Programmers-Book.pdf 5. UML для бізнес-моделювання: для чого потрібні діаграми процесів. URL: https://evergreens.com.ua/ua/articles/uml-diagrams.html 6. Простий посібник зі схем UML і моделювання баз даних. URL: https://www.microsoft.com/uk-ua/microsoft-365/business-insights-ideas/resources/guide-to-uml-diagramming-and-database-modeling 7. Розділ 2. Основи UML. URL: https://docs.kde.org/trunk5/uk/umbrello/umbrello/uml-basics.html 8. Стандарт UML 2.2. URL: http://www.omg.org/technology/documents/modeling_spec_catalog.htm#UML
Тривалість курсу	90 год.
Обсяг курсу	32 години аудиторних занять. З них 16 годин лекцій, 16 годин лабораторних робіт занять та 58 годин самостійної роботи.
Компетентності та програмні результати навчання	<p>При вивченні дисципліни «Об'єктно-орієнтований дизайн» здобувачі вищої освіти набувають такі компетентності (здатність):</p> <p>ІК1 – Здатність визначати та розв'язувати складні економічні задачі та проблеми, приймати відповідні аналітичні та управлінські рішення у сфері економіки або у процесі навчання, що передбачає проведення досліджень та/або здійснення інновацій за невизначених умов та вимог.</p> <p>ЗК1 – Здатність генерувати нові ідеї (креативність).</p> <p>ЗК4 – Здатність спілкуватися з представниками інших професійних груп різного рівня (з експертами з інших галузей знань/видів економічної діяльності).</p> <p>ЗК5 – Здатність працювати в команді.</p> <p>ЗК6 – Здатність розробляти та управляти проектами.</p> <p>СК4 – Здатність використовувати сучасні інформаційні технології, методи та прийоми дослідження економічних та соціальних процесів, адекватні встановленим потребам дослідження.</p> <p>СК11 – Здатність планувати і розробляти проекти у сфері економіки, здійснювати її інформаційне, методичне, матеріальне, фінансове та кадрове забезпечення.</p> <p>СК12 – 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>СК14 – Здатність формувати нові конкурентоспроможні ідеї й реалізовувати їх у проектах (стартапах).</p> <p>СК15 – Здатність самостійно опановувати нові знання, використовуючи сучасні освітні та дослідницькі технології у сфері економіки й інформаційних технологій та приймати ефективні управлінські рішення.</p>

	<p>Програмні результати навчання:</p> <p>ПР03 – Вільно спілкуватися з професійних та наукових питань державною та іноземною мовами усно і письмово.</p> <p>ПР06 – Оцінювати результати власної роботи, демонструвати лідерські навички та вміння управляти персоналом і працювати в команді.</p> <p>ПР17 – Керувати технологічними процесами ІТ виробництва та здійснювати контроль якості програмної продукції.</p> <p>ПР18 – Виявляти основні тенденції в ІТ-сфері та бізнес-перспективи використання конкретних інформаційних технологій в умовах динамічності ринкового середовища.</p> <p>ПР19 – Розробляти управлінські рішення щодо забезпечення розвитку державних, муніципальних, комерційних організацій та бізнес-структур в усіх галузях економіки через призму інформаційних технологій; усвідомлювати необхідність навчання впродовж усього життя з метою поглиблення набутих та здобуття нових фахових знань.</p> <p>Після завершення цього курсу студент буде :</p> <p>а) знати</p> <ul style="list-style-type: none"> • сучасні засоби моделювання і документування архітектури ПЗ; • методи побудови та аналізу формальних моделей предметних областей; • основні фази процесу розробки ПЗ; • основи мови UML; • принципи об'єктно-орієнтованого проектування; • принципи і методи створення складних програмних засобів на основі концепції та стандартів відкритих систем, CASE-систем, мови UML; • функціональні можливості однієї з програм-редакторів мови UML; • основні архітектурні шаблони, які використовуються при розробці програмних систем; • шаблони проектування, що використовуються при розробці систем; <p>б) уміти</p> <ul style="list-style-type: none"> • розробляти моделі компонентів інформаційних систем, включаючи моделі баз даних; • обґрунтовувати прийняті проектні рішення, здійснювати постановку і виконувати експерименти з перевірки їх коректності та ефективності; • моделювати, аналізувати та документувати вимоги до коректності та ефективності ПЗ; • методи і засоби організації колективної роботи над проектом по розробці ПЗ; • застосовувати на практиці фундаментальні концепції об'єктно-орієнтованого підходу: абстракцію, інкапсуляцію, успадкування, поліморфізм; • проводити аналіз вимог і проектувати, на основі даних аналізу, системи і підсистеми з використанням мови моделювання UML; • розробляти логічну модель програмної системи; • розробляти компонентну модель програмної системи; • розробляти модель розміщення компонентів програмної системи; • застосовувати шаблони проектування та архітектурні шаблони.
<p>Ключові слова</p>	<p>Об'єктно-орієнтоване програмування, об'єктно-орієнтований аналіз, об'єктно-орієнтований дизайн, мова UML, об'єкти, класи, життєвий цикл, колаборація, аналітична модель, UML-діаграми, шаблони, метрики, об'єктно-реляційне відображення.</p>

Формат курсу	Очний
	Проведення лекцій, лабораторних робіт та консультації для кращого розуміння тем. Викладання навчальної дисципліни передбачає поєднання традиційних форм аудиторного навчання з елементами електронного навчання, в якому використовуються спеціальні інформаційні технології, такі як комп'ютерна графіка, аудіо та відео, інтерактивні елементи, онлайн консультування і т.п.
Теми	Тема 1. Еволюція методологій програмування Тема 2. Об'єкти та класи Тема 3. Об'єктно-орієнтований аналіз Тема 4. Принципи об'єктно-орієнтованого дизайну Тема 5. Уніфікована мова моделювання UML Тема 6. Об'єктно-реляційне відображення Тема 7. Шаблони проектування Тема 8. Метрики
Підсумковий контроль, форма	Залік в кінці семестру/комбінований (відповідь + письмовий тест). Оцінка складається із кількості балів нарахованих за: здачу лабораторних робіт, виконання самостійних робіт та індивідуального завдання, написання контрольної модульної роботи. Методи контролю: спостереження за навчальною діяльністю здобувачів вищої освіти, усне опитування, письмовий контроль, тестовий контроль, виконання навчальних та індивідуальних завдань.
Пререквізити	Для вивчення курсу студенти потребують базових знань з дисциплін як "Технології створення програмних та інтелектуальних систем", "Проектний менеджмент", "ІТ-стратегії бізнесу", "Ефективність інформаційних систем", "Сучасні інформаційні технології" достатніх для: а) сприйняття категоріального апарату сучасної сфери інформаційних технологій та розробки програмних засобів; б) оперування методами професійної дискусії для формування власної аргументованої позиції; в) формування аналітичного та проектного мислення; г) формування навичок моделювання бізнес-процесів мовою UML.
Навчальні методи та техніки, які будуть використовуватися під час викладання курсу	Презентація, лекція-бесіда, лекція-візуалізація, колаборативне навчання (форми – групові проекти, спільні розробки і т. д.), проектно-орієнтоване навчання, навчальна дискусія, мозкова атака, кейс-метод, демонстрування, самостійна робота, лабораторні роботи, метод порівняння, метод узагальнення, метод конкретизації, метод виокремлення основного, обговорення, робота над помилками,
Необхідне обладнання	Вивчення курсу потребує використання програмного забезпечення як от Rational Rose, редактори UML, Office 365, крім загально вживаних програм і операційних систем. Мультимедійна дошка, проектор.
Критерії оцінювання (окремо для кожного виду навчальної діяльності)	Оцінювання проводиться за 100-бальною шкалою. Бали нараховуються за наступним співвідношенням: <ul style="list-style-type: none"> • лабораторні роботи: 36% семестрової оцінки; максимальна кількість балів – 36; • індивідуальне науково-дослідне завдання: 14% семестрової оцінки; максимальна кількість балів –14; • самостійна робота: 10% семестрової оцінки; максимальна кількість балів – 10; • контрольна робота + модульна контрольна робота: 40% семестрової оцінки (максимальна кількість балів – 40).

	<p>Підсумкова максимальна кількість балів – 100.</p> <p>Політика щодо дедлайнів та перескладання: роботи, які здаються із порушенням термінів без поважних причин, оцінюються на нижчу оцінку – 20% від визначених балів за даний вид роботи. Студенти виконують декілька видів письмових робіт (есе, підготовка рефератів). Перескладання (модулів, контрольних робіт) відбувається із дозволу лектора за наявності поважних причин (наприклад, довідка про стан здоров'я).</p> <p>Академічна доброчесність: 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 практичній (письмовій) 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 зайняття курсу. Студенти мають інформувати викладача про неможливість відвідати заняття. У будь-якому випадку студенти зобов'язані дотримуватися усіх строків 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 запізень 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>Жодні форми порушення академічної доброчесності не толеруються.</p>
<p>Питання до заліку</p>	<ol style="list-style-type: none"> 1. Покоління мов програмування. 2. Об'єктна модель. 3. Об'єктні мови програмування. 4. Об'єктно-орієнтовані мови. 5. Об'єктно-орієнтований аналіз, дизайн та проектування. 6. Парадигми програмування. 7. Складові частини об'єктного підходу. 8. Абстрагування. 9. Інкапсуляція. 10. Модульність. 11. Ієрархія. 12. Типізація. 13. Паралелізм. 14. Збережуваність. 15. Поняття об'єкту в ООП. 16. Стан та поведінка об'єкту. Ідентичність. 17. Життєвий цикл об'єкту. 18. Відношення між об'єктами.

19. Взаємодія клієнт-сервер.
20. Ієрархія об'єктів.
21. Структура класів.
22. Абстрактну класи, інтерфейси та класи-утиліти.
23. Відношення між класами: асоціація, агрегація та композиція, використання, наслідування.
24. Інстанціювання.
25. Цілі і завдання фази ООА.
26. Процес переходу від фази збору вимог до фази ООА.
27. Аналітична модель.
28. Аналітичні класи і відносини між ними.
29. Реалізації варіантів використання.
30. Колаборації.
31. Опис колаборацій за допомогою діаграм взаємодії.
32. Принципи дизайну класів.
33. Правило цілісності абстракції.
34. Закон Деметри.
35. Принцип закритості.
36. Принцип підстановки (Liskov).
37. Принцип розділення інтерфейсів.
38. Принципи проектування залежностей.
39. Принцип інверсії залежності.
40. Принцип ациклічності залежностей.
41. Принципи проектування пакетів.
42. Загальний принцип повторного використання.
43. Принцип локалізації змін.
44. Принцип стабільності залежностей.
45. Принцип стабільних абстракцій. Абстрактність.
46. Генеральна послідовність.
47. Мова UML.
48. Основні функції UML.
49. Загальна структура мови UML.
50. Типи схем UML.
51. Базові семантичні конструкції мови, їх опис за допомогою спеціальних позначень.
52. Типи відношень в UML.
53. Механізми розширення UML.
54. Види діаграм в UML.
55. Діаграма класів.
56. Діаграма компонентів.
57. Діаграма композитної / складеної структури.
58. Діаграма кооперації (UML2.0).
59. Діаграма розгортання.
60. Діаграма об'єктів.
61. Діаграма пакетів.
62. Діаграма діяльності.
63. Діаграма станів (або скінченних автоматів).
64. Діаграма прецедентів (або варіантів використання).
65. Діаграма кооперації (UML1.x) / комунікації (UML2.0).
66. Діаграма огляду взаємодії (UML2.0).
67. Діаграма послідовності.
68. Діаграма синхронізації (UML2.0).
69. Особливості графічного зображення діаграм мови UML.

	<ul style="list-style-type: none"> 70. Пакети в UML. 71. Редактор UML-діаграм. 72. Об'єктно-орієнтовані БД. 73. Об'єктно-реляційна БД. 74. Що таке ORM? 75. Парадигма «невідповідності». 76. Принцип роботи ORM. 77. Переваги і недоліки використання ORM. 78. Визначення, класифікація та опис шаблонів проектування. 79. Архітектурні шаблони. 80. Породжуючі шаблони. 81. Структурні шаблони. 82. Шаблони поведінки. 83. Кількісні метрики. 84. Метрики складності управління програмами. 85. Метрики складності потоку даних програми. 86. Комбіновані метрики складності управління і даних. 87. Об'єктно-орієнтовані метрики і метрики надійності. 88. Гібридні метрики.
<p>Дотримання умов доброчесності</p>	<p>Академічна доброчесність: Очікується, що роботи студентів в розділі ІНДР 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 не доброчесності. 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. В Університеті наявні документи, які містять політику, стандарти і процедури дотримання академічної доброчесності: Положення про систему внутрішнього забезпечення якості освіти ЛНУ; Положення про забезпечення академічної доброчесності у Львівському національному університеті імені Івана Франка; Кодекс академічної доброчесності Львівського національного університету імені Івана Франка» (https://cutt.ly/ofX2uIH); Декларація про дотримання академічної доброчесності працівником у Львівському національному університеті імені Івана Франка (https://cutt.ly/nfX2itr); Декларація про дотримання академічної доброчесності здобувачем вищої освіти у Львівському національному університеті імені Івана Франка (https://cutt.ly/4fX2iE6). Дані документи розміщені за посиланням https://education-quality.lnu.edu.ua</p>
<p>Опитування</p>	<p>Анкету-оцінку з метою оцінювання якості курсу буде надано по завершенню курсу.</p>

Схема курсу

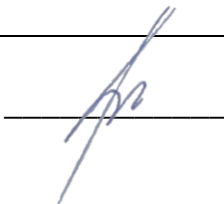
Тиждень год.	Тема, план, короткі тези	Форма діяльності (заняття)	Література. Ресурси в інтернеті	Завдання, год.	Термін виконання
1	2	3	4	5	6
Тиж. 1 2 год.	<p>Тема 1. Еволюція методологій програмування Покоління мов програмування. Об'єктна модель. Об'єктні мови програмування. Об'єктно-орієнтовані мови. Об'єктно-орієнтований аналіз, дизайн та проектування. Парадигми програмування. Складові частини об'єктного підходу. Абстрагування. Інкапсуляція. Модульність. Ієрархія. Типізація. Паралелізм. Збережувальність</p> <p>Тема 2. Об'єкти та класи Поняття об'єкту в ООП. Стан та поведінка об'єкту. Ідентичність. Життєвий цикл об'єкту. Відношення між об'єктами. Взаємодія клієнт-сервер. Ієрархія об'єктів. Структура класів. Абстрактну класи, інтерфейси та класи-утиліти. Відношення між класами: асоціація, агрегація та композиція, використання, наслідування. Інстанціювання.</p>	Лекція	Осн. [1-16]. Дод. [1-7]. Інт. [4-7].	Опрацювати лекційний матеріал, підготуватися до лабораторного заняття 2 год.	До проведення наступного аудиторного заняття за розкладом
Тиж. 2 2 год.	<p>Тема 1. Еволюція методологій програмування Тема 2. Об'єкти та класи Дослідження предметної області. Створення глосарію.</p>	Лабораторна робота	Осн. [1-16]. Дод. [1-7]. Інт. [4-7].	Обговорення проблемних питань, виконання лабораторної роботи, 2 год.	До проведення наступного аудиторного заняття за розкладом
Тиж. 3 2 год.	<p>Тема 3. Об'єктно-орієнтований аналіз Цілі і завдання фази аналізу. Процес переходу від фази збору вимог до фази аналізу. Аналітична модель. Аналітичні класи і відносини між ними. Реалізації варіантів використання. Колаборації. Опис колаборацій за допомогою діаграм взаємодії.</p>	Лекція	Осн. [1-16]. Дод. [1-7]. Інт. [1-10].	Опрацювати лекційний матеріал, підготуватися до лабораторного заняття 2 год.	До проведення наступного аудиторного заняття за розкладом

1	2	3	4	5	6
Тиж. 4 2 год.	Тема 3. Об'єктно-орієнтований аналіз Визначення функціональних та нефункціональних вимог до програмного продукту, виділення дійових осіб та їх інтересів.	Лабораторна робота	Осн. [1-16]. Дод. [1-7]. Інт. [1-10].	Обговорення проблемних питань, виконання лабораторної роботи, 2 год.	До проведення наступного аудиторного заняття за розкладом
Тиж. 5 2 год.	Тема 4. Принципи об'єктно-орієнтованого дизайну Принципи дизайну класів. Правило цілісності абстракції. Закон Деметри. Принцип закритості. Принцип підстановки (Liskov). Принцип розділення інтерфейсів. Принципи проектування залежностей. Принцип інверсії залежності. Принцип ациклічності залежностей. Принципи проектування пакетів. Загальний принцип повторного використання. Принцип локалізації змін. Принцип стабільності залежностей. Стабільність. Принцип стабільних абстракцій. Абстрактність. Генеральна послідовність.	Лекція	Осн. [1-16]. Дод. [1-7]. Інт. [1-10].	Опрацювати лекційний матеріал, підготуватися до лабораторного заняття, 2 год.	До проведення наступного аудиторного заняття за розкладом
Тиж. 6 2 год.	Тема 4. Принципи об'єктно-орієнтованого дизайну Складання описів варіантів використання	Лабораторна робота	Осн. [1-16]. Дод. [1-7]. Інт. [1-10].	Обговорення проблемних питань, виконання лабораторної роботи, 2 год.	До проведення наступного аудиторного заняття за розкладом
Тиж. 7 2 год.	Тема 5. Уніфікована мова моделювання UML Мова UML. Основні функції UML. Загальна структура мови UML. Типи схем UML. Базові семантичні конструкції мови, їх опис за допомогою спеціальних позначень. Типи відношень в UML. Механізми розширення UML. Види діаграм в UML. Діаграма класів. Діаграма компонентів. Діаграма композитної / складеної структури. Діаграма кооперації (UML2.0). Діаграма розгортання. Діаграма об'єктів.	Лекція	Осн. [1, 4-8, 13-16]. Дод. [1, 3]. Інт. [8, 10].	Опрацювати лекційний матеріал, підготуватися до лабораторного заняття 2 год.	До проведення наступного аудиторного заняття за розкладом
Тиж. 8 2 год.	Тема 5. Уніфікована мова моделювання UML Діаграма класів. Діаграма об'єктів	Лабораторна робота	Осн. [1, 4-8, 13-16].	Обговорення проблемних питань,	До проведення

1	2	3	4	5	6
			Дод. [1, 3]. Інт. [8, 10].	виконання лабораторної роботи, 2 год.	наступного аудиторного заняття за розкладом
Тиж. 9 2 год.	Тема 5. Уніфікована мова моделювання UML Діаграма пакетів. Діаграма діяльності. Діаграма станів (або скінченних автоматів). Діаграма прецедентів (або варіантів використання). Діаграма кооперації (UML1.x) / комунікації (UML2.0). Діаграма огляду взаємодії (UML2.0). Діаграма послідовності. Діаграма синхронізації (UML2.0). Особливості графічного зображення діаграм мови UML. Пакети в UML. Редактор UML-діаграм.	Лекція	Осн. [1, 4-8, 13-16]. Дод. [1, 3]. Інт. [8, 10].	Опрацювати лекційний матеріал, підготуватися до лабораторного заняття 2 год.	До проведення наступного аудиторного заняття за розкладом
Тиж. 10 2 год.	Тема 5. Уніфікована мова моделювання UML Діаграма використання	Лабораторна робота	Осн. [1, 4-8, 13-16]. Дод. [1, 3]. Інт. [8, 10].	Обговорення проблемних питань, виконання лабораторної роботи, 2 год.	До проведення наступного аудиторного заняття за розкладом
Тиж. 11 2 год.	Тема 6. Об'єктно-реляційне відображення Об'єктно-орієнтовані БД. Об'єктно-реляційна БД. Що таке ORM? Парадигма «невідповідності». Принцип роботи ORM. Переваги і недоліки використання. Список програм об'єктно-реляційного відображення.	Лекція	Осн. [1-16]. Дод. [1-7]. Інт. [1-10].	Опрацювати лекційний матеріал, підготуватися до лабораторного заняття 2 год.	До проведення наступного аудиторного заняття за розкладом
Тиж. 12 2 год.	Тема 5. Уніфікована мова моделювання UML Діаграма послідовності	Лабораторна робота	Осн. [1, 4-8, 13-16]. Дод. [1, 3]. Інт. [8, 10].	Обговорення проблемних питань, виконання лабораторної роботи, 2 год.	Згідно розкладу
Тиж. 13 2 год.	Тема 7. Шаблони проектування	Лекція	Осн. [1-16]. Дод. [1-7]. Інт. [9].	Опрацювати лекційний матеріал,	До проведення наступного

1	2	3	4	5	6
	Визначення, класифікація та опис шаблонів проектування. Архітектурні шаблони. Породжуючі шаблони. Структурні шаблони. Шаблони поведінки.			підготуватися до лабораторного заняття, 2 год.	аудиторного заняття за розкладом
Тиж. 14 2 год.	Тема 5. Уніфікована мова моделювання UML Діаграма діяльності	Лабораторна робота	Осн. [1, 4-8, 13-16]. Дод. [1, 3]. Інт. [8, 10].	Обговорення проблемних питань, виконання лабораторної роботи, 2 год.	До проведення наступного аудиторного заняття за розкладом
Тиж. 15 2 год.	Тема 8. Метрики Кількісні метрики. Метрики складності управління програмами. Метрики складності потоку даних програми. Комбіновані метрики складності управління і даних. Об'єктно-орієнтовані метрики і метрики надійності. Гібридні метрики.	Лекція	Осн. [1-16]. Дод. [1-7]. Інт. [9].	Опрацювати лекційний матеріал, підготуватися до лабораторного заняття, 2 год.	До проведення наступного аудиторного заняття за розкладом
Тиж. 16 2 год.	Підсумковий контроль	Тестування	Осн. [1-16]. Дод. [1-7]. Інт. [1-10].	Виконання індивідуальних практичних завдань, тестових завдань	Згідно розкладу

Викладач



Ірина ШЕВЧУК