

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

Оновлено та затверджено
на засіданні
кафедри прикладної математики
факультету прикладної математики та
інформатики
Львівського національного університету
імені Івана Франка
(протокол № 5 від 19.11 2023 р.)



Завідувач кафедри

Юрій ЯЦУК

Силабус з навчальної дисципліни
«Проектування об'єктно-зорієнтованих програм»,
що викладається в межах ОНП Прикладна математика
другого (магістерського) рівня вищої освіти для здобувачів
з спеціальності 113 – прикладна математика

Львів 2023 р.

Назва дисципліни	Проектування об'єктно-зорієнтованих програм
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 прикладної математики
Галузь знань, шифр та назва спеціальності	11 Математика та статистика 113 Прикладна математика
Викладачі дисципліни	Дяконюк Лілія Миколаївна, доцент кафедри прикладної математики, Марчук Юрій Богданович, асистент кафедри прикладної математики.
Контактна інформація викладачів	Liliiia.Diakoniuk@lnu.edu.ua ; https://ami.lnu.edu.ua/employee/diakoniuk ; Yurii.Marchuk@lnu.edu.ua , https://ami.lnu.edu.ua/employee/marchuk Головний корпус ЛНУ ім. І. Франка, каб. 262, 361. м. Львів, вул. Університетська, 1
Консультації з питань навчання по дисципліні відбуваються	Консультації в день проведення лекцій/лабораторних занять (за попередньою домовленістю).
Сторінка курсу	https://ami.lnu.edu.ua/course/proektuvannia-ob-iektno-zoriientovanykh-program-trykkladna-matematyka-1-9
Інформація про дисципліну	Дисципліна «Проектування об'єктно-зорієнтованих програм» є дисципліною за вибором студента з спеціальності 113 – прикладна математика для ОНП «Прикладна математика», яка викладається в 2-му семестрі (4,5 кредити ECTS).
Коротка анотація дисципліни	Курс передбачає вивчення основних принципів проектування програмного забезпечення для великих програмних комплексів з використанням об'єктно-зорієнтованої парадигми, порівняння переваг різних парадигм, а також стандартних шаблонів проектування.
Мета та цілі дисципліни	Метою вивчення вибіркової дисципліни «Проектування об'єктно-зорієнтованих програм» є освоєння студентами основних принципів розробки програмних проектів на об'єктно-зорієнтованій мові, створення архітектури програмного забезпечення з використанням шаблонів проектування, яка б давала можливість ефективно вносити зміни в структуру, уникати повторів коду, а також ускладнень, економно використовувати ресурси системи та забезпечувати надійність програмного забезпечення.
Література для вивчення дисципліни	Основна література 1. Erich Gamma. Design Patterns: Elements of Reusable Object-Oriented Software" / Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides (Gamma of Gang of Four). - 632 p. 2. E. Freeman. Head First Design Patterns 2E. / E. Freeman, E.Robson. - O'Reilly Media -2020. - 641p.

	<p>3. Jeffrey Chilberto. Hands-On Design Patterns with C# and .NET Core. / Jeffrey Chilberto, Gaurav Arora. - ISBN-139781789133646. – Packt Publishing. – 2019. – 410 p.3.</p> <p>4. A Shvetc. Design patterns. – Refactoring.guru, 2021. - 410 p.</p> <p>5. D. Boswell. The Art of Readable Code: Simple and Practical Techniques for Writing Better Code. / D. Boswell, T.Foucher. - ISBN-9780596802295 (ISBN10: 0596802293). - O'Reilly Media. - 2011. – 204 p.</p> <p>Допоміжна: Р. Мартін. Чиста архітектура. - ISBN-978-617-09-5286-8. – Фабула. – 2022. – 832с.</p>
Обсяг курсу	Загальний обсяг: 135 годин. Аудиторних занять: 64 год., з них 16 год. лекційних та 32 години лабораторних робіт. Самостійної роботи: 87 год.
Очікувані результати навчання	<p>Після завершення цього курсу студент буде</p> <p>Знати:</p> <ul style="list-style-type: none"> - Основні парадигми ОЗП - Переваги і недоліки принципів Solid - Мову UML. - Шаблони проектування - Пастки програмування. <p>Вміти:</p> <ul style="list-style-type: none"> - Будувати різні види UMLдіаграм. - Обґрунтовувати вибрану схему архітектури програмного забезпечення - Використовувати шаблони проектування - Уникати стандартних пасток в проектуванні, які приводять до проблемних ситуацій. <p>Реалізовувати шаблони на обраній мові, з врахуванням специфіки реалізації</p>
Ключові слова	Паралельні комп'ютери, багатоядерність, багатопроцесорність, паралельні обчислення, схеми з'єднань, граф алгоритму, шарова схема зберігання.
Формат курсу	Очний. Проведення лекцій, лабораторних занять і консультацій.
Теми	Подано нижче у таблиці Схема курсу «Проектування об'єктно-зорієнтованих програм»
Підсумковий контроль, форма	Залік
Пререквізити	Для вивчення курсу студенти потребують базових знань з основ програмування, дисциплін “Програмне забезпечення”, “Об'єктно-орієнтоване програмування”.
Навчальні методи та техніки, які будуть використовуватися під час викладання курсу	Презентації, лекції (лекція-розповідь, лекція-бесіда). Індивідуальні завдання, командний проєкт.

Необхідне об-ладнання	Комп'ютер із програмним забезпеченням, що дозволяють реалізовувати програми на мовах, обраних студентами (C++, C#, Java, Python), доступ до Internet мережі.				
Критерії оцінювання (окремо для кожного виду навчальної діяльності)	Оцінювання проводиться за 100-бальною шкалою.				
	Оцінка за шкалою ECTS		Оцінка в балах	Оцінка за національною шкалою	
			залік		
	A	Відмінно	100 - 90	зараховано	
	B	Дуже добре	81- 89		
	C	Добре	71 -80		
	D	Задовільно	61 - 70		
E	Достатньо	51- 60			
FX (F)	Незадовільно	0 - 50	не зараховано		
Поточне оцінювання: впродовж семестру студент може отримати 100 балів. З них:					
- за роботу на лабораторних заняттях: максимальна кількість – 48 балів.					
Виконання проєкту: максимальна кількість – 52 балів.					
Підсумкове оцінювання проводиться у вигляді заліку.					
Загалом протягом семестру 100 балів.					
Критерії оцінювання завдання наведено у додатку.					
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 письмовій роботі студента є підставою для її незарахування викладачем, незалежно від масштабів плагіату чи обману.					
Відвідання занять 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 письмових робіт та індивідуальних завдань, передбачених курсом.					
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.					
Політика виставлення балів. Враховуються бали набрані на лабораторних заняттях (здача завдань), При цьому обов'язково враховуються робота на заняттях та активність студента під час лабораторного заняття; недопустимість пропусків та запізнь на заняття; списування та плагіат; несвоєчасне виконання поставленого завдання і т. ін.					

	Жодні форми порушення академічної доброчесності не толеруються.
Опитування	Анкету-оцінку з метою оцінювання якості курсу буде надано по завершенню курсу.

Схема курсу «Проектування об'єктно-зорієнтованих програм»

Тиж.	Тема, план, короткі тези	Форма діяльності (заняття)	Література, Ресурси в інтернеті	Завдання, год.	Термін виконання
1	Тема 1. Огляд курсу. Особливості розробки великих програмних комплексів.	Лекція (2 год.)	[1,3,4,6,7]	Опрацювання лекційного матеріалу (2 год.)	1 тиждень
2	Тема 2. Огляд парадигм програмування з визначенням місця ОЗП.	Лекція (2 год.)	[1-5,7,11,12]	Опрацювання лекційного матеріалу (2 год.)	1 тиждень
	Тема 1-2. Розв'язування задач з демонстрацією переваг та недоліків різних підходів. Обговорення особливостей групових проектів. Розподіл підзадач для виконання.	Лабораторне (4 год)	[1-5,7,11,12]	Опрацювання матеріалу (4 год.)	2 тижні
3	Тема 3. Мова UML. Типи діаграм та особливості їх використання.	Лекція (2 год.)	[5,7,8]	Опрацювання лекційного матеріалу (4 год.)	2 тижні
	Тема 3. Побудова діаграм взаємозв'язків між класами, діаграми послідовностей викликів та діаграми прецедентів	Лабораторне (4 год)	[1-5,7,11,12]	Створення діаграм для групового проекту (8 год)	2 тижні
4	Тема 4. Класифікація шаблонів проектування. Принципи SOLID. Основне призначення шаблонів проектування.	Лекція (2 год.)	[5,8]	Опрацювання лекційного матеріалу (4 год)	2 тижні
	Тема 4. Розробка програм з використанням принципів SOLID. Огляд можливих суперечностей та наслідків порушень.	Лабораторне (4 год.)	[1-5,7,11,12]	Виконання домашнього завдання 1 (4 год.) Створення класів для групового проекту (6 год)	2 тижні

5	Тема 5. Породжуючі шаблони.	Лекція (2 год.)	[5,7,8]	Опрацювання лекційного матеріалу (4 год.)	2 тижні
6	Тема 6. Реалізація шаблонів Одинак, Абстрактна фабрика, Фабричний метод	Лабораторне (4 год.)	[5,8,9-12]	Виконання домашнього завдання, (4 год.) Робота над груповим проектом (6 год.)	2 тижні
7	Тема 7. Поведінкові шаблони.	Лекція, (2 год.)	[1-5,7,11,12]	Опрацювання лекційного матеріалу (4 год.)	2 тижні
	Тема 7. Розв'язування задач з використанням шаблонів Ланцюг обов'язків, Відвідувач, Посередник, спостерігач.	Лабораторне (4 год.)	[2,9-12]	Виконання домашнього завдання, (4 год.) робота над груповим проектом (6 год.)	2 тижні
8.	Тема 8. Структурні шаблони.	Лекція (2 год.)	[5,7,8]	Опрацювання лекційного матеріалу (4 год.)	2 тижні
	Тема 8. Робота з шаблонами Декоратор, Фасад, Міст. Особливості поєднання різних шаблонів.	Лабораторне (6 год.)	[5,7,8]	Виконання домашнього завдання, (4 год.), робота над груповим проектом. (5 год.)	3 тижні
9	Тема 9. Презентація групових проєктів. Обговорення переваг і недоліків обраної методики проєктування.	Лекція (2 год.)	[1-4,6]	Підготовка звіту. Огляд стандартних пасток проєктування. (4 год.)	2 тижні
	Тема 9. Захист групових проєктів.	Лабораторне (4 год.)		Аналіз виконання роботи іншої команди(8 год)	2 тижні

Оцінювання проводиться за 100-бальною шкалою

Оцінка за шкалою ECTS		Оцінка в балах	Оцінка за національною шкалою
			залік
A	Відмінно	100 - 90	зараховано
B	Дуже добре	81- 89	
C	Добре	71 -80	
D	Задовільно	61 - 70	
E	Достатньо	51- 60	
FX(F)	Незадовільно	0 - 50	не зараховано

Розподіл балів, які отримують студенти:

Поточний контроль:

Лабораторні заняття – 48 балів (3x16)

Самостійна робота – 52 балів

Підсумковий контроль:

Бали нараховуються за наступним співвідношенням

Лабораторні заняття – максимальна кількість балів – 48 б.

(16 занять - максимально 3 б. за заняття)

3 бали – студент в повному обсязі володіє навчальним матеріалом;

2 бали – студент достатньо володіє навчальним матеріалом;

1 бал – студент задовільно володіє навчальним матеріалом ;

0 балів – невиконання завдань

Самостійна робота || Індивідуальна робота

- навчально-науковий командний проєкт і презентація його результатів – максимальна кількість балів – 52 б.

Для виставлення оцінок за різними критеріями застосовується шкала балів:

- побудова діаграм проєкту 10 б
- обґрунтованість застосування шаблонів проєктування: 10 б
- застосування принципів SOLID, обґрунтування 10 б
- програмна реалізація: 10 б
- презентація і захист результатів на семінарському занятті: 12 б.

Деталізація оцінювання:

Побудова діаграм проєкту 10 б:

10 балів – діаграми побудовано у повному обсязі, правильно відображено всі елементи та взаємозв'язки, студент впевнено і докладно відповідає на поставлені запитання під час захисту і демонструє розуміння всіх конструкцій;

9-8 балів – діаграми побудовано у повному обсязі з незначними помилками або неточностями, студент майже впевнено відповідає на поставлені запитання;

7-5 балів – діаграми побудовано не у повному обсязі, проте основні конструкції відображено правильно, студент не завжди дає вичерпні відповіді на запитання;

4-1 балів – діаграми побудовано не у повному обсязі, є помилки у відображенні основних конструкцій, під час захисту студент не може відповісти на жодне поставлене запитання.

0 балів – діаграми не побудовані.

Обґрунтованість застосування шаблонів проєктування: 10 б:

10 балів: студент правильно розуміє шаблони, які використовує для проєктування, знає альтернативні можливості, вміє проаналізувати переваги і недоліки запропонованого підходу, розуміє, які особливості програмної реалізації запропонованих шаблонів.

9-6 балів: студент правильно розуміє використані шаблони, проте не всі умови задачі є виконаними, студентом не проаналізовано альтернативні варіанти розв'язку.

5-3 балів: шаблони проєктування не обґрунтовано не використані, проте є використані абстрактні конструкції.

3-1 балів. Студент не використовує абстракцій, але є розроблені класи для відображення основних сутностей.

0 балів – завдання повністю не виконано.

Застосування принципів SOLID, обґрунтування 10 б:

10 балів: студент для всіх побудованих класів може вказати, які принципи SOLID були використані при побудові, може пояснити, які з принципів є порушеними і обґрунтувати, якщо їх неможливо реалізувати. При захисті студент може навести приклади застосувань та порушень по кожному з принципів.

9-5 балів: студент не у всіх випадках розуміє принципи використання, не всі принципи може проаналізувати, проте дає визначення принципів.

5-1 бал: студент вміє сформулювати не всі визначення принципів, частково неправильно їх застосовує, або необґрунтовано їх не застосовує.

Програмна реалізація: 10 б

10 балів: завдання виконано повністю, програмний код відповідає проєкту, або при програмній реалізації внесено зміни в проєкт, які повністю обґрунтовані. Студент повністю вміє пояснити всі особливості реалізації коду.

9-7 балів: завдання виконане, програмний код відповідає проєкту, проте не ефективно реалізовані окремі методи. Студент повністю вміє пояснити всі особливості реалізації коду.

6-5 балів: завдання виконане, проте програмний код не завжди відповідає проєкту, а внесені зміни не обґрунтовані. Студент повністю вміє пояснити всі особливості реалізації коду.

4-3 балів: завдання виконане не повністю, програмний код не завжди відповідає проєкту, а внесені зміни не обґрунтовані. Студент повністю вміє пояснити всі особливості реалізації коду.

2 -1 балів: завдання виконане не повністю, програмний код не завжди відповідає проєкту, а внесені зміни не обґрунтовані. Програма не повністю працює. Студент не повністю вміє пояснити всі особливості реалізації коду.

0 – не реалізовано програмний продукт.

Презентація і захист результатів на семінарському занятті: 12 б.

12 балів: створена презентація, матеріал викладено в повному обсязі, послідовно, використані наглядні способи візуалізації. Студент володіє термінами. Правильно відповідає на поставлені колегами та викладачем запитання.

11-6 балів: створена презентація, матеріал викладено в повному обсязі, послідовно, використані наглядні способи візуалізації. Студент не всі терміни використовує правильно. Не всі відповіді на запитання є обґрунтовані.

5-1 балів: створена презентація не в повному обсязі відображає опрацьований матеріал. Студент не може відповісти на поставлені запитання.

Додаткові бали можна отримати за наступні активності:

Написання тез, статті, участь у міжнародних, всеукраїнських та/або інших заходах або конкурсах за тематикою