

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ВІДОКРЕМЛЕНИЙ СТРУКТУРНИЙ ПІДРОЗДІЛ
«ФАХОВИЙ КОЛЕДЖ ПРОМИСЛОВОЇ АВТОМАТИКИ
ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
ОДЕСЬКОГО НАЦІОНАЛЬНОГО ТЕХНОЛОГІЧНОГО УНІВЕРСИТЕТУ»**

Циклова комісія комп'ютерних наук та інженерії програмного забезпечення

ЗАТВЕРДЖУЮ

Заступник директора

з навчально-методичної роботи

_____Ірина ГЕНЕРАЛОВА

__._.2025 року

**РОБОЧА ПРОГРАМА З НАВЧАЛЬНОЇ ДИСЦИПЛІНИ
КОНСТРУЮВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ**

обов'язкова

Освітньо-професійна програма «Комп'ютерні науки»_____

Код та найменування спеціальності 122 Комп'ютерні науки_____

Шифр та найменування галузі знань 12 Інформаційні технології_____

Мова навчання українська

Розроблено та забезпечується: цикловою комісією Комп'ютерних наук та інженерії програмного забезпечення ВСП «Фаховий коледж промислової автоматики та інформаційних технологій Одеського національного технологічного університету»

Розробник:

Юлія БУРМАКІНА, викладач вищої кваліфікаційної категорії ФКПАІТ ОНТУ

Розглянуто та схвалено на засіданні циклової комісії Комп'ютерних наук та інженерії програмного забезпечення

Протокол № 1 від 27. 08. 2025 р.

Голова циклової комісії

(підпис)

Тетяна КОСТИРЕНКО

(Власне ім'я, ПРІЗВИЩЕ)

Гарант освітньо-професійної програми

(підпис)

Наталія ШВЕЦЬ

(Власне ім'я, ПРІЗВИЩЕ)

Розглянуто та схвалено Методичною радою ФКПАІТ ОНТУ

Протокол №__ від __.__.20__р.

Голова Методичної ради ФКПАІТ ОНТУ

(підпис)

Ірина ГЕНЕРАЛОВА

(Власне ім'я, ПРІЗВИЩЕ)

1. Пояснювальна записка

Мета та завдання навчальної дисципліни

Метою викладання навчальної дисципліни «Конструювання програмного забезпечення» є формування системи теоретичних знань і набуття практичних умінь і навичок з питань теоретичних основ та використання сучасних методологій та технологій розробки, тестування та експлуатації програмних продуктів, та для моделювання і проектування інформаційних систем, документування та маркетингу програмного забезпечення. Для моделювання та проектування розглядається уніфікована мова моделювання UML.

Основними завданнями вивчення дисципліни «Конструювання програмного забезпечення» є:

1. **Засвоєння основ UML**, зокрема опанування базових понять та елементів уніфікованої мови моделювання, таких як діаграми варіантів використання, діаграми класів, діаграми послідовностей та інші.
2. **Моделювання вимог засобами UML**, що передбачає застосування діаграм варіантів використання для формалізованого та наочного опису функціональних вимог до програмного продукту.
3. **Проектування програмних систем із використанням діаграм класів і співпраці**, що дозволяє визначати структуру системи та характер взаємодії між її компонентами.
4. **Моделювання сценаріїв взаємодії**, шляхом використання діаграм послідовностей для відображення обміну повідомленнями між об'єктами системи в різних сценаріях її функціонування.
5. **Опис поведінки системи та її компонентів**, через застосування діаграм станів для представлення переходів між різними станами об'єктів.
6. **Використання UML для документування програмних проєктів**, з метою створення структурованої, зрозумілої та повної технічної документації.
7. **Застосування UML у практичній проєктній діяльності**, що передбачає реалізацію набутих знань під час розроблення навчальних програмних проєктів.

Метою реалізації зазначених завдань є формування у здобувачів освіти навичок застосування UML як дієвого засобу аналізу, проектування та документування програмного забезпечення, що забезпечує підвищення ефективності командної взаємодії та створення якісних програмних продуктів у практичних умовах сучасної IT-індустрії.

Необхідним елементом успішного засвоєння навчального матеріалу дисципліни є самостійна робота студентів з літературою з питань базисної технології створення програмних продуктів як сукупності процесів розроблення програмних систем.

Компетентності та результати навчання

У результаті вивчення навчальної дисципліни «Конструювання програмного забезпечення» здобувач освіти отримує наступні програмні компетентності та програмні результати навчання, які визначені в Стандарті фахової передвищої освіти із спеціальності 122 «Комп'ютерні науки» (<https://mon.gov.ua/static-objects/mon/sites/1/Fakhova%20peredvyshcha%20osvita/Zatverdzheni.standarty/2021/11/30/122-Kompyuterni.nauky.30.11.pdf>) та освітньо-професійній програмі

«Комп'ютерні науки»

(<https://promavt.od.ua/storage/uploads/DXbHAM7yoQmTpSH4pc5TUAPTakBhyuPfTSBhLInL.pdf>) підготовки фахових молодших бакалаврів.

Загальні компетентності:

ЗК03. Здатність до абстрактного мислення, аналізу та синтезу.

ЗК04. Здатність застосовувати знання в практичних ситуаціях.

ЗК05. Знання та розуміння предметної області та розуміння професійної діяльності.

ЗК08. Здатність вчитися і оволодівати сучасними знаннями.

Спеціальні (фахові, предметні) компетентності:

СК02. Здатність використовувати теоретичні та фундаментальні знання в галузі комп'ютерних наук та інформаційних технологій для вирішення різноманітних проблем.

СК03. Здатність розробляти, аналізувати та застосовувати ефективні алгоритми для розв'язання конкретних професійних задач залежно від предметного середовища.

СК04. Здатність здійснювати проектування та розробку програмного забезпечення.

СК08. Здатність застосовувати сучасні методи, технології та інструментальні засоби проектування та створення програмних систем та їх супроводження.

СК11. Здатність застосовувати методи та техніки тестування програмного забезпечення впродовж життєвого циклу розробки програмних систем.

СК14. Здатність проектувати ІС у відповідності до профілю підготовки по видах забезпечення.

Програмні результати навчання:

РН10. Знати методології, методи, моделі, процеси і технології життєвого циклу розробки та тестування програмного забезпечення.

РН12. Знати основні принципи функціонування системного та прикладного програмного забезпечення.

РН15. Розробляти супровідну документацію на різних етапах процесу життєвого циклу розробки програмного забезпечення.

Міждисциплінарні зв'язки

Попередні - «Основи програмної інженерії», «Об'єктно-орієнтоване програмування», «Бази даних»; послідовні – «Тестування програмного забезпечення», «Технології захисту інформації».

2. Опис навчальної дисципліни

[illegible]

3. Зміст навчальної дисципліни

№ з/п	Назви змістових модулів і тем , зміст заняття	Кількість годин			
		Усього	У тому числі		
			Лекції	Практичні	Самостійна робота
	Змістовий модуль 1. Конструювання програмного забезпечення.				
1	Вступ до конструювання програмного забезпечення. Задачі конструювання програмного забезпечення. Поняття конструювання програмного забезпечення (КПЗ). Головні компоненти КПЗ. Область знань КПЗ. Стилi КПЗ. Моделі конструювання, планування та внесення змін.		2		6
2	Планування конструювання програмного забезпечення Постановка завдання. Визначення основних термінів (потреби, вимога, формалізація постановки завдання, аналіз вимог, специфікація, управління вимогами). Планування конструювання.		2		8
3	Проблеми розробки складних програмних систем Фактори, що збільшують складність розробки програмних систем. Блочно-ієрархічний підхід конструювання програмних систем. Декомпозиція складних об'єктів. Покрокова деталізація. Локальна оптимізація. Рівні та аспекти проектування. Процеси технологічного циклу конструювання ПЗ. Методи керування і планування проектом.		2		8
4	Проектування при конструюванні. Проблеми, пов'язані з проектуванням програмного забезпечення. Бажані характеристики проекту.				6
5	Проектування при конструюванні. Рівні проектування. Компоненти проектування.				6
6	Проектування при конструюванні. Методики проектування.				6
7	Життєвий цикл програмного забезпечення. Поняття життєвого циклу програмного забезпечення. Стадії створення програмного забезпечення.		2		8
8	Моделі життєвого циклу програмного забезпечення.		2		8

	Поняття моделі життєвого циклу програмного забезпечення. Каскадна, ітеративна та спіральна моделі життєвого циклу програмного забезпечення.				
	Разом за змістовим модулем 1	66	10		56
	Змістовий модуль 2: Уніфікована мова моделювання UML.				
1	Мова моделювання UML. Структура і компоненти мови UML. Загальні принципи моделювання систем. Словник UML. Предмети. Поняття <i>сутності</i> в UML. Види сутностей в UML. Поняття <i>класу, інтерфейсу, кооперації, прецеденту, компоненту</i> та <i>вузла</i> . Відношення в UML. Поняття <i>діаграми</i> в UML.		2		6
2	Основи об'єктно-орієнтованого представлення програмних систем. Об'єкти. Загальна характеристика. Поняття <i>декомпозиції</i> . Принципи об'єктно-орієнтованої декомпозиції: абстрагування, інкапсуляція, модульність й ієрархічна організація. Загальна характеристика об'єктів. Поняття об'єкт, індивідуальність, стан, поведінка. Види операцій. Види відношень між об'єктами: зв'язок та агрегація.		2		6
3	Практична робота №1 Опис та аналіз предметної області. Навчитися виконувати опис предметної області проєктованої програмної системи. Проводити об'єктний аналіз отриманого опису і будувати модель середовища за допомогою діаграми потоків даних (аналіз поведінки системи) і діаграми «сутність-зв'язок» (аналіз даних). Навчитися визначати призначення проєктованої ІКС.			2	
4	Діаграма прецедентів (діаграма варіантів використання або Use case Diagram). Діаграма прецедентів. Актори та варіанти використання на діаграмі прецедентів. Інтерфейси та примітки на діаграмі прецедентів.		2		8
5	Відношення між елементами на діаграмі прецедентів. Зв'язки між елементами діаграми прецедентів: відношення асоціації, відношення узагальнення, відношення включення, відношення розширення. Графічна нотація діаграми. Навики побудови діаграми за вибраною предметною областю.		2		8
6	Практична робота №2 Побудова діаграми прецедентів (Use case Diagram) Отримати практичні навички побудови діаграм Use Case.			4	

7	Діаграма класів. Діаграма класів. Переваги та недоліки класів. Поняття класу. Компоненти діаграми класів. Елементи діаграми класів: класи, інтерфейси, залежності, узагальнення та асоціації. Поняття імені, атрибуту та операції(методу) класу. Квантор видимості та значення, які він може приймати.		2		8
8	Відношення між класами на діаграмі класів. Відношення між класами: асоціація, залежність, узагальнення, реалізація, агрегація, композиція. Графічна нотація діаграми. Навики побудови діаграми за вибраною предметною областю.		2		8
9	Практична робота №3 Побудова діаграми класів (Class Diagram) Отримати практичні навички побудови діаграми класів.			4	
10	Діаграма схем станів. Моделювання поведінки програмної системи. Поняття <i>автомату</i> та <i>взаємодії</i>. Призначення діаграми схем станів. Обов'язкові умови формалізму звичайного автомату. Поняття стану. Зображення стану на діаграмі. Секції, на які може бути розподілений стан: ім'я стану та список внутрішніх дій або переходів. Поняття <i>мітки дій</i> та значення, які вони можуть приймати. Початковий та кінцевий стан на діаграмі станів. Графічна нотація діаграми. Навички побудови діаграми за вибраною предметною областю.		4		8
11	Практична робота №4 Побудова діаграми схем станів (Statechart Diagram) Отримати практичні навички побудови діаграми станів.			4	
12	Діаграма діяльності. Різниця діаграми діяльності від діаграми станів. Елементи, які використовуються на діаграмі діяльності. Поняття <i>діяльності</i> та <i>стану</i> на діаграмі діяльності. Переходи на діаграмі. Призначення доріжок на діаграмі діяльності. Об'єкти на діаграмі. Графічна нотація діаграми. Навички побудови діаграми за вибраною предметною областю.		2		8
13	Практична робота №5 Побудова діаграми діяльності (Activity Diagram) Отримати практичні навички побудови діаграми діяльності.			4	
14	Діаграма послідовностей дій. Призначення діаграми послідовностей дій. Об'єкти на діаграмі. Призначення понять <i>лінія життя</i> та <i>фокус керування</i>. Види повідомлень. Графічна нотація діаграми. Навички побудови діаграми.		2		8

15	Практична робота №6 Побудова діаграми послідовності дій. Отримати практичні навички побудови діаграми послідовності.			4	
16	Діаграма співпраці (кооперацій). Цілі діаграми співпраці. Базові поняття. Кооперація. Об'єкти на діаграмі. Графічне зображення об'єктів на діаграмі кооперації. Стереотипи зв'язків. Повідомлення. Формат запису повідомлень. Практичні приклади побудови діаграм кооперації		2		8
17	Практична робота №7 Побудова діаграми співпраці (кооперації) (Collaboration Diagram) Отримати практичні навички побудови діаграми співпраці.			4	
18	Діаграма компонентів. Діаграма компонентів. Елементи діаграми компонентів: компоненти та інтерфейси. Різновиди компонентів. Особливості компонентів. Стандартні стереотипи, передбачені в UML для компонентів. Графічна нотація діаграми. Навички побудови діаграми за вибраною предметною областю.		2		6
19	Практична робота №8 Побудова діаграми компонентів (Component Diagram) Отримати практичні навички побудови діаграми компонентів.			2	
20	Діаграма розгортання. Діаграма розгортання(розміщення). Елементи діаграми розгортання: вузли, відношення залежності, відношення асоціації. Графічна нотація діаграми. Навички побудови діаграми за вибраною предметною областю.		2		6
21	Практична робота №9 Побудова діаграми розгортання (Deployment Diagram) Отримати практичні навички побудови діаграми розгортання.			2	
	Разом за змістовим модулем 2	144	26	30	88
	Разом з дисципліни	210	36	30	144

**4. Критерії оцінювання результатів навчання
для дисциплін освітньо-професійної програми**

Критерії оцінювання навчальних досягнень здобувача освіти		Значення оцінки результатів навчання за формами підсумкового контролю		
Рівень досягнення здобувачем освіти запланованих результатів навчання з навчальної дисципліни	Рівень сформованості компетентностей, визначених освітньо-професійною програмою для навчальної дисципліни	Екзамен/ диференційований залік		
		Оцінка за національною шкалою	Оцінка за 12-бальною шкалою	Оцінка за 100-бальною шкалою
Здобувач освіти демонструє високий рівень досягнення запланованих результатів вивчення навчальної дисципліни, що засвідчують його безумовну готовність до подальшого навчання та/або професійної діяльності за фахом	Здобувач освіти виявляє високий рівень сформованості всіх загальних і фахових компетентностей, передбачених освітньо-професійною програмою спеціальності	відмінно	12	90-100
			11	
			10	
Здобувач освіти виявляє достатній рівень досягнення запланованих результатів вивчення навчальної дисципліни та готовності до подальшого навчання та/або професійної діяльності за фахом	Здобувач освіти демонструє достатній рівень сформованості загальних і фахових компетентностей, передбачених освітньо-професійною програмою спеціальності. Несформовані компетентності відсутні	добре	9	74-89
			8	
			7	

Наявні мінімально достатні для подальшого навчання та/або професійної діяльності за фахом результати вивчення навчальної дисципліни	Здобувач освіти демонструє мінімально достатній та достатній рівень сформованості загальних і фахових компетентностей, передбачених освітньо-професійною програмою спеціальності. Несформовані компетентності відсутні	задовільно	6	60-73
			5	
			4	
Ряд запланованих результатів вивчення навчальної дисципліни відсутній. Рівень наявних результатів навчання є недостатнім є недостатнім для подальшого навчання та/або професійної діяльності за фахом	Лише частина визначених освітньо-професійною програмою спеціальності для навчальної дисципліни загальних і фахових компетентностей сформовані у здобувача освіти на мінімально-достатньому або недостатньому рівні, рівень сформованості решти компетентностей є недостатнім або компетентність взагалі відсутній	незадовільно (з можливістю перескладання)	3	40-59
			2	
Результати навчання відсутні	Компетентності не сформовані	незадовільно (з необхідністю повторного вивчення дисципліни)	1	0-39

Схема нарахування балів, які отримують здобувачі освіти з навчальної дисципліни

Вид діяльності здобувача освіти	Кількість балів		Кількість робіт	Сумарні бали	
	min	max		min	max
Змістовий модуль 1: Конструювання програмного забезпечення.					
Відвідування лекцій	2	2	5	10	10
Виконання завдань для самостійної роботи	1	2	1	1	2
Виконання контрольної роботи/тестування	1,5	3	1	1,5	3
Оцінка за змістовий модуль 1				12,5	15
Змістовий модуль 2: Уніфікована мова моделювання UML.					
Відвідування лекцій	2	2	13	26	26
Практична робота (в тому числі допуск, виконання, захист)	2	6	9	18	54
Виконання завдань для самостійної роботи	1	2	1	1	2
Виконання контрольної роботи/тестування	1,5	3	1	1,5	3
Оцінка за змістовий модуль 2				46,5	85
Разом за семестр				60	100

5. Засоби діагностики результатів навчання

Перевірка та оцінювання знань здобувачів освіти може проводитись кількома методами:

1. Оцінювання знань здобувача освіти під час практичних занять.
2. Виконання індивідуальних навчально-дослідних завдань.
3. Захист практичних робіт.
4. Тестування.
5. Проведення поточно-модульного контролю.
6. Проведення заліку.

6. Інструменти, обладнання та програмне забезпечення, використання яких передбачає дисципліна

1. Безкоштовний онлайн-інструмент для створення діаграм, блок-схем та візуальних схем - <https://app.diagrams.net/>
2. Інструмент для візуального моделювання програмного забезпечення - <https://staruml.io/>
3. Інструмент для побудови UML-схем - <https://creately.com>

7. Інформаційні ресурси

Базові (основні):

- Григорович О.Є., Ковальчук В.В. «Моделювання програмних систем засобами UML» - Львів : Видавництво Львівської політехніки, 2012.
- Кунгурцев О.Б. «Конструювання програмного забезпечення. Об'єктно-орієнтований підхід» - навчальний посібник
- Павленко П.М., Федорчук В.А. «UML у проектуванні програмного забезпечення» - К.: НАУ, 2010.
- Жежерун С.М. «Об'єктно-орієнтований аналіз та проектування з використанням UML» - Тернопіль : ТНТУ, 2014.

Додаткова література

- С. О. Цибульник, К. С. Барандич «Технології розроблення програмного забезпечення частина 1. Життєвий цикл програмного забезпечення», Київ КПІ ім. Ігоря Сікорського 2022
- <http://www.znannya.org> – електронний ресурс
- <http://sites.znu.edu.ua> – електронний ресурс