



Силабус освітнього компонента

Програма навчальної дисципліни



Об'єктно-орієнтоване програмування

Шифр та назва спеціальності

121 – Інженерія програмного забезпечення

Інститут

ННІ Комп'ютерних наук та інформаційних технологій

Освітня програма

Інженерія програмного забезпечення

Кафедра

Програмна інженерія та інтелектуальні технології управління (321)

Рівень освіти

Бакалавр

Тип дисципліни

Спеціальна (фахова), Обов'язкова

Семестр

3

Мова викладання

Українська, англійська

Викладачі, розробники



Іванов Лев Вадимович

lev.ivanov@khpi.edu.ua

старший викладач кафедри програмної інженерії та інтелектуальних технологій управління

Google Scholar: <https://scholar.google.com/citations?user=ADPHLAsAAAAJ>

[Детальніше про викладача на сайті кафедри](#)

Загальна інформація

Анотація

Завданням дисципліни є засвоєння студентами необхідного рівня знань щодо складових частин об'єктно-орієнтованої парадигми, об'єктно-орієнтованої моделі та синтаксису мови об'єктно-орієнтованого програмування C#, а також застосування об'єктно-орієнтованого підходу, методів і техніки створення програм з застосуванням мови об'єктно-орієнтованого програмування C#.

Мета та цілі дисципліни

Засвоєння необхідних знань з опанування сучасними технологіями аналізу, проектування та об'єктно-орієнтованої розробки програмних систем.

Формат занять

Лекції, лабораторні роботи, самостійна робота, консультації. Підсумковий контроль – іспит.

Компетентності

K01. Здатність до абстрактного мислення, аналізу та синтезу.

K02. Здатність застосовувати знання у практичних ситуаціях.

K05. Здатність вчитися і оволодівати сучасними знаннями.

K06. Здатність до пошуку, оброблення та аналізу інформації з різних джерел.

K07. Здатність працювати в команді.

K14. Здатність брати участь у проектуванні програмного забезпечення, включаючи проведення моделювання (формальний опис) його структури, поведінки та процесів функціонування.

- K15. Здатність розробляти архітектури, модулі та компоненти програмних систем.
- K22. Здатність накопичувати, обробляти та систематизувати професійні знання щодо створення і супроводження програмного забезпечення та визнання важливості навчання протягом всього життя.
- K23. Здатність реалізовувати фази та ітерації життєвого циклу програмних систем та інформаційних технологій на основі відповідних моделей і підходів розробки програмного забезпечення.
- K25. Здатність обґрунтовано обирати та освоювати інструментарій з розробки та супроводження програмного забезпечення.

Результати навчання

- ПР01. Аналізувати, цілеспрямовано шукати і вибирати необхідні для вирішення професійних завдань інформаційно-довідникові ресурси і знання з урахуванням сучасних досягнень науки і техніки.
- ПР04. Знати і застосовувати професійні стандарти і інші нормативно-правові документи в галузі інженерії програмного забезпечення.
- ПР05. Знати і застосовувати відповідні математичні поняття, методи доменного, системного і об'єктно-орієнтованого аналізу та математичного моделювання для розробки програмного забезпечення.
- ПР07. Знати і застосовувати на практиці фундаментальні концепції, парадигми і основні принципи функціонування мовних, інструментальних і обчислювальних засобів інженерії програмного забезпечення.
- ПР08. Вміти розробляти людино-машинний інтерфейс.
- ПР13. Знати і застосовувати методи розробки алгоритмів, конструювання програмного забезпечення та структур даних і знань.
- ПР14. Застосовувати на практиці інструментальні програмні засоби доменного аналізу, проектування, тестування, візуалізації, вимірювань та документування програмного забезпечення.
- ПР16. Мати навички командної розробки, погодження, оформлення і випуску всіх видів програмної документації.
- ПР17. Вміти застосовувати методи компонентної розробки програмного забезпечення.
- ПР23. Вміти документувати та презентувати результати розробки програмного забезпечення.

Обсяг дисципліни

Загальний обсяг дисципліни 120 год. (4 кредити ECTS): лекції – 32 год., лабораторні роботи – 32 год., самостійна робота – 56 год.

Передумови вивчення дисципліни (пререквізити)

Основи програмування
Теорія алгоритмів
Ознайомча практика в "Innovation Campus"

Особливості дисципліни, методи та технології навчання

Методи викладання та навчання:

інтерактивні лекції з презентаціями, дискусії, лабораторні заняття, командна робота, кейс-метод, метод зворотного зв'язку з боку студентів, проблемне навчання.

Форми оцінювання:

письмові індивідуальні завдання до лабораторних робіт (CAS), оцінювання знань на лабораторних заняттях (CAS), експрес-опитування (CAS), онлайн-тести (CAS), підсумковий/семестровий контроль у формі семестрового екзамену, відповідно до графіку навчального процесу (FAS).

Програма навчальної дисципліни

Теми лекційних занять

Тема 1. Основи ООП. Базовий синтаксис мови С#. Типи-посилання

Парадигми програмування. Переваги об'єктно-орієнтованого підходу. Складові об'єктно-орієнтованої методології. Платформа .NET та мова програмування С#. Базовий синтаксис мови С#. Простори імен. Класи. Властивості. Робота з масивами. Індексатори. Рядки. Консольне виведення та введення.

Тема 2. Додаткові можливості класів. Успадкування та поліморфізм. Структури та переліки

Ініціалізатори об'єктів. Статичні класи. Перевантаження операцій. Успадкування. Поліморфізм. Інтерфейси. Безіменні типи і записи. Структури і переліки. Кортежі. Методи, які розширюють існуючі класи. Вкладені типи. Зіставлення зі зразком.

Тема 3. Робота з винятками та файлами. Узагальнення та колекції.

Обробка виняткових ситуацій. Початкові відомості про роботу з файлами. Робота з XML-документами. Використання серіалізації. Створення та використання узагальнень. Стандартні узагальнені класи та методи. Створення власних контейнерних типів. Створення та використання бібліотек класів.

Тема 4. Неповні класи. Подіє-орієнтоване, функційне та декларативне програмування

Тип dynamic. Неповні класи та методи. Робота з файловою системою. Делегати. Події. Концепції функційного та декларативного програмування. Реалізація мовою С# функційного та декларативного програмування. Розробка GUI-застосунків за допомогою технології Windows Presentation Foundation.

Тема 5. Застосування патернів проектування в С#

агальні концепції патернів проектування. Класифікація патернів проектування. Приклади твірних патернів. Приклади структурних патернів. Приклади патернів поведінки. Розв'язання задач проектування за допомогою патернів.

Теми практичних занять

Практичні заняття в рамках дисципліни не передбачені.

Теми лабораторних робіт

Тема 1. Основи ООП. Базовий синтаксис мови С#

Тема 2. Успадкування та поліморфізм

Тема 3. Узагальнене програмування

Тема 4. Подіє-орієнтоване, функційне та декларативне програмування

Тема 5. Патерни проектування

Самостійна робота

Навчальним планом передбачено виконання курсової роботи (КР). На початку семестру студенти обирають теми КР з переліку або пропонують власні теми та погоджують їх з викладачем. КР виконується протягом семестру та захищається на заліковому тижні або екзаменаційні сесії. Студентам рекомендуються додаткові матеріали для самостійного вивчення та опрацювання.

Література та навчальні матеріали

Основна література

1. Troelsen A. Japikse P. Pro C# 9 with .NET 5: Foundational Principles and Practices in Programming: 10th edition, Apress, 2021, 1411 p.
2. Albahari J. C# 9.0 in a Nutshell: The Definitive Reference: 1st Edition, O'Reilly Media, 2021, 1060 p.
3. Nagel C. Professional C# 7 and .NET Core 2.0: 7th Edition, Wrox, 2018, 1440 p.
4. Booch G. Rumbaugh J, Jacobson I. The Unified Modeling Language User Guide (Object Technology Series): 2nd Edition, Addison-Wesley Professional, 2005, 494 p.
5. Gamma E., Helm R., Johnson R., Vlissides J. Design Patterns: Elements of Reusable Object-Oriented Software: 1st Edition, Addison-Wesley Professional, 1994, 540 p.

Додаткова література

1. Farrell J. Microsoft Visual C#: An Introduction to Object-Oriented Programming: 007 Edition, Joyce Cengage Learning; 007 edition, 2017, 784 p.
2. Cardoso A. F. M. Implementing Design Patterns in C# and .NET 5: Build Scalable, Fast, and Reliable .NET Applications Using the Most Common Design Patterns. BPB Publications, 2021, 290 p.
3. Weisfeld M. Object-Oriented Thought Process, The Developer's Library: 5th Edition, 2019, 240 p.

Система оцінювання

Критерії оцінювання успішності студента та розподіл балів

100% підсумкової оцінки складаються з результатів оцінювання у вигляді екзамену (40%) та поточного оцінювання (60%):

- 5 лабораторних робіт (по 5%);
- курсова робота (35%).

Шкала оцінювання

Сума балів	Національна оцінка	ECTS
90–100	Відмінно	A
82–89	Добре	B
75–81	Добре	C
64–74	Задовільно	D
60–63	Задовільно	E
35–59	Незадовільно (потрібне додаткове вивчення)	FX
1–34	Незадовільно (потрібне повторне вивчення)	F

Норми академічної етики і політика курсу

Студент повинен дотримуватися «Кодексу етики академічних взаємовідносин та доброчесності НТУ «ХПІ»: виявляти дисциплінованість, вихованість, доброзичливість, чесність, відповідальність. Конфліктні ситуації повинні відкрито обговорюватися в навчальних групах з викладачем, а при неможливості вирішення конфлікту – доводитися до відома співробітників дирекції інституту. Нормативно-правове забезпечення впровадження принципів академічної доброчесності НТУ «ХПІ» розміщено на сайті: <http://blogs.kpi.kharkov.ua/v2/nv/akademichna-dobrochesnist/>

Погодження

Силабус погоджено

11.04.2023

Завідувач кафедри
Ігор ГАМАЮН

08.06.2023

Гарант ОП
Юлія ЛІТВІНОВА