



Відділення інженерії програмного забезпечення
Циклова комісія програмування

СИЛАБУС

Базова інформація про дисципліну	
Назва дисципліни	Об'єктно-орієнтоване програмування
Рівень вищої освіти / фахової передвищої освіти	Фахова передвища
Галузь знань	12 «Інформаційні технології»
Спеціальність	121 «Інженерія програмного забезпечення»
Освітня програма	121 «Інженерія програмного забезпечення»
Семестр	5,6 семестр (9 кл), 3,4 семестр (11 кл)
Факультет / відділення	Програмування
Курс	3 курс (9 кл), 2 курс (11 кл)
Анотація курсу	Вивчення сукупності прийомів об'єктно-орієнтованого програмування при розробці клієнтських додатків, можливостей мови програмування C#, розробка Windows-додатків з графічним інтерфейсом та веб-додатків засобами платформи .NET
Сторінка курсу в MOODLE	http://78.137.2.119:1919/m72/course/view.php?id=106
Мова викладання	Українська
Лектор курсу	Викладач II категорії Марченко С.В. СДН «Moodle»: повідомлення в чаті E-mail: numosmanius@gmail.com
Місце дисципліни в освітній програмі	
Освітня програма	http://csbc.edu.ua/documents/otdel/moop_pr.pdf
Перелік загальних компетентностей (ЗК)	Знання та розуміння предметної області та розуміння професійної діяльності. Здатність до пошуку, оброблення та аналізу інформації з різних джерел. Здатність до абстрактного мислення, аналізу та синтезу. Здатність застосовувати знання у практичних

	ситуаціях.
<p>Перелік спеціальних компетентностей (СК)</p>	<p>Здатність накопичувати, обробляти та систематизувати професійні знання щодо створення і супроводу програмного забезпечення та визнання важливості навчання протягом усього життя.</p> <p>Здатність застосовувати фундаментальні та міждисциплінарні знання для успішного розв'язання завдань інженерії програмного забезпечення.</p> <p>Здатність дотримуватися специфікацій, стандартів, правил і рекомендацій в професійній галузі при реалізації процесів життєвого циклу програмного забезпечення.</p> <p>Здатність брати участь у проектуванні програмного забезпечення, включаючи проведення моделювання (формальний опис) його структури, поведінки та процесів функціонування.</p> <p>Здатність обґрунтовано обирати та освоювати інструментальні засоби розробки та супроводу програмного продукту.</p> <p>Здатність розробляти архітектури, модулі та компоненти програмного забезпечення</p>
<p>Перелік програмних результатів навчання</p>	<p>Вміти застосовувати методи обчислення та структури даних для вирішення задач аналізу та синтезу алгоритмів.</p> <p>Вміти систематизувати та узагальнювати інформацію про підходи, методи та засоби розробки супроводу програмного забезпечення, вдосконалювати відповідні знання, вміння й навички протягом усього життя.</p> <p>Вміти знаходити аналогії та застосовувати знання, вміння та навички з суміжних дисциплін для формування та вирішення професійних завдань.</p> <p>Знати і застосовувати методи доменного, системного і об'єктно-орієнтованого аналізу для розробки програмного забезпечення.</p> <p>Знати основні методології та підходи до розробки програмного забезпечення, вміти обирати та реалізовувати модель життєвого циклу програмного забезпечення.</p> <p>Знати основні стандарти у галузі програмної інженерії та вміти дотримуватись рекомендацій, стандартів, специфікацій стосовно процесів</p>

	<p>життєвого циклу програмного забезпечення.</p> <p>Знати основні стандарти, принципи та нотації проектування програмного забезпечення та вміти застосовувати їх на практиці для моделювання структури, поведінки та внутрішніх процесів програмної системи.</p> <p>Мати навички командної розробки, погодження, оформлення і випуску всіх видів програмної документації.</p> <p>Знати основні методи оптимізації алгоритмів, вміти розробляти ефективні алгоритми розв'язування завдань та на їх основі створювати програмний код.</p> <p>Знати основні інструментальні засоби для розробки та супроводу програмного забезпечення та вміти застосовувати їх на практиці з урахуванням специфіки отриманого завдання та вимог користувача.</p> <p>Знати основні типи архітектури програмного забезпечення, стандартні архітектурні рішення (патерни проектування) та вміти застосовувати їх під час проектування архітектури складних багатокомпонентних програмних систем.</p>
Опис дисципліни	
<p>Структура навантаження на студента</p>	<p>Загальна кількість годин – 180</p> <p>Кількість кредитів – 6</p> <p>Кількість лекційних годин – 47</p> <p>Кількість практичних занять – 60</p> <p>Кількість годин для самостійної роботи студентів – 73</p> <p>Форма підсумкового контролю – залік/ екзамен</p>
<p>Методи навчання</p>	<p>За подачею навчального матеріалу: методи готових знань, дослідницький метод.</p> <p>З огляду на мету навчання: методи здобуття нових знань, метод формування умінь і навичок, метод застосування знань на практиці, методи закріплення знань, умінь і навичок, методи перевірки і оцінювання знань, умінь і навичок.</p>

Зміст дисципліни	
Тема 1. Вступ до об'єктно-орієнтованого програмування	Базові концепції об'єктно-орієнтованого програмування. Огляд платформи .NET. Структура програми мовою C#. Система типів мови програмування C#
Тема 2. Знайомство з базовим синтаксисом мови програмування C#	Управляючі інструкції мови програмування C#. Посилальні числові типи даних. Робота з масивами. Програмні засоби для роботи з текстом. Регулярні вирази
Тема 3. Фундаментальні принципи ООП. Інкапсуляція	Базова робота з класами. Інкапсуляція та приховування даних у мові C#. Конструювання об'єктів. Принцип відокремлення команд та запитів. Принцип надійності Постела
Тема 4. Фундаментальні принципи ООП. Взаємодія класів	Види взаємодії класів. Наслідування на прикладі структурованої обробки винятків. Поліморфізм та його різновиди. Перетворення типів та поліморфізм підтипів
Тема 5. Фундаментальні принципи ООП. Абстрагування даних та параметричний поліморфізм	Абстрактні класи. Інтерфейси. Вбудовані інтерфейси .NET. Делегати та анонімні типи. Параметричний поліморфізм. Узагальнені типи даних
Тема 6. Принципи побудови якісного об'єктно-орієнтованого коду	Філософія об'єктно-орієнтованого проектування програмного забезпечення. SOLID-принципи розробки об'єктно-орієнтованого коду. Принципи написання модульних тестів. Створення тестових наборів за допомогою MS Test v2
Тема 7. Колекції в мові програмування C#	Огляд API колекцій. Стандартні структури даних на платформі .NET. Доступні тільки для читання та спостережувані колекції
Тема 8. Файловий ввід-вивід у мові програмування C#	Файловий ввід-вивід. Потоки вводу-виводу. Серіалізація об'єктів
Тема 9. Організація доступу до даних за допомогою технології LINQ	Огляд технології LINQ. Фільтрування, впорядкування та проектування даних. З'єднання, групування та агрегування даних
Тема 10. Асинхронне та паралельне виконання	Загальні відомості про конкурентне виконання коду. Асинхронні операції на базі синтаксису

коду засобами мови C#	async-await. Основи паралельного виконання коду. Паралельна обробка даних за допомогою Parallel LINQ
Тема 11. Технологія Windows Presentation Foundation	Декларативна мова програмування XAML. Технологія Windows Presentation Foundation. Компонування інтерфейсу за допомогою контейнерних елементів. Основні елементи управління XAML. Взаємодія з додатком за допомогою подій
Тема 12. Огляд MVVM-архітектури додатків з графічним інтерфейсом	Принципи роботи архітектури MVVM. Формування рівня моделі та взаємодія з ним. Поняття ресурсів у WPF-додатках. Технологія прив'язування даних
Тема 13. Розробка розвиненого інтерфейсу користувача засобами WPF	Стилізація елементів управління. Декорування та шаблонізація елементів управління. Система анімацій у WPF
Тема 14. Основи роботи з технологією Entity Framework Core	Огляд архітектури ASP.NET Core. Створення першого веб додатку. Обробка запитів за допомогою middleware-конвеєра
Тема 15. Розробка вебсайту на основі ASP.NET Core MVC	Створення веб сторінок з MVC-контролерами. Відображення URL на методи за допомогою маршрутизації. Модель прив'язування даних. Рендеринг HTML з використанням Razor-представлень. Побудова форм з використанням тег хелперів
Тема 16. Розробка односторінкового вебсайту з використанням UI-фреймворку Blazor	Технології Blazor Server та Blazor WebAssembly. Створення односторінкового веб сайту. Побудова Blazor-бібліотеки для повторного використання. Розгортання Blazor-додатків
Політика дисципліни	
Політика відвідування	Регулярне відвідування всіх видів занять, своєчасність виконання самостійної роботи. За об'єктивних причин (наприклад, хвороба, міжнародне стажування) навчання зорганізується в он-лайн формі за погодженням із керівником курсу.
Політика щодо дедлайнів та	Роботи, які здаються із порушенням термінів без поважних причин, оцінюються на нижчу оцінку.

перескладання	
Академічна доброчесність	У випадку недотримання політики академічної доброчесності (плагіат, самоплагіат, фабрикація, фальсифікація, списування, обман, хабарництво) передбачено повторне проходження оцінювання.

Система оцінювання

Поточний контроль здійснюється протягом семестру під час проведення практичних, семінарських та інших видів занять і оцінюється сумою набраних балів (максимальна сума – 100 балів; мінімальна сума, що дозволяє студенту отримати атестацію з предмету – 60 балів); підсумковий/ семестровий контроль, проводиться у формі заліку або іспиту, відповідно до графіку навчального процесу.

Підсумкова оцінка за умови заліку виставляється як загальна сума балів, набраних за результатами поточного контролю. Підсумкова оцінка за умови іспиту виставляється як загальна сума балів набраних за результатами поточного (70%) та підсумкового контролю.

Накопичування рейтингових балів з навчальної дисципліни

5-й семестр

Види навчальної роботи	Мах кількість балів
Практичні завдання (10 тем)	50
Тестування (4 теми)	20
Індивідуальне практичне завдання	30
Разом	100

6-й семестр

Види навчальної роботи	Мах кількість балів
Практичні завдання (8 тем)	24
Тестування (4 теми)	16
Індивідуальне практичне завдання	30
Екзамен	30
Разом	100

Шкала оцінювання

ECTS	Бали	Зміст
A	90-100	Бездоганна підготовка в широкому контексті
B	80-89	Повні знання, міцні вміння
C	70-79	Хороші знання та вміння
D	65-69	Задовільні знання, стереотипні

		вміння
E	60-64	Виконання мінімальних вимог діяльності в стандартних умовах
FX	35-59	Слабкі знання, відсутність умінь
F	1-34	Необхідний повторний курс

Список рекомендованих джерел

1. Коноваленко І.В., Марущак П.О., Савків В.Б. Програмування мовою С# 7.0: навчальний посібник. Тернопіль: Тернопільський національний технічний університет імені Івана Пулюя, 2017. 300 с.
2. Троелсен, Э., Джепикс, Ф. Язык программирования С# 7 и платформы .NET и .NET Core, 8-е изд.: Пер. с англ. СПб.: ООО “Диалектика”, 2018, 1328 с.: ил. Парал. тит. англ.
3. Клири С. Конкурентность в С#. Асинхронное, параллельное и многопоточное программирование. 2-е межд. изд. СПб.: Питер, 2020. 272 с.: ил. (Серия «Для профессионалов»).
4. Дібрівний О. А., Гребенюк В. В. Вступ до об'єктно-орієнтованого програмування С#: Навчальний посібник / Київ: Державний університет телекомунікацій, 2018, - 190с.