

 <p>ЧЕРКАСЬКИЙ ДЕРЖАВНИЙ БІЗНЕС-КОЛЕДЖ</p>	<p>Кафедра комп'ютерної інженерії та інформаційних технологій</p> <p><b>СИЛАБУС</b></p>
<p><b>Базова інформація про дисципліну</b></p>	
<p><b>Назва дисципліни</b></p>	<p><b>SE004 Об'єктно-орієнтоване програмування Object-Oriented Programming</b></p>
<p><b>Рівень вищої освіти / фахової передвищої освіти</b></p>	<p>перший (бакалаврський)</p>
<p><b>Галузь знань</b></p>	<p>12 Інформаційні технології</p>
<p><b>Спеціальність</b></p>	<p>123 «Комп'ютерна інженерія»</p>
<p><b>Освітня програма</b></p>	<p>Комп'ютерна інженерія</p>
<p><b>Семестр</b></p>	<p>1 семестр</p>
<p><b>Курс</b></p>	<p>1 курс</p>
<p><b>Анотація курсу</b></p>	<p>Курс охоплює основи об'єктно-орієнтованого аналізу, проектування програмного забезпечення, а також об'єктно-орієнтованого програмування на мові Java.</p> <p>Зокрема розглядається термінологія і фундаментальні концепції об'єктно-орієнтованого програмування такі як абстракція, інкапсуляція, наслідування і поліморфізм.</p> <p>Студентам даються практичні навички об'єктно-орієнтованого аналізу предметної області, визначення архітектури системи з виявленням варіантів використання системи, моделюванням класів і взаємодій між ними використовуючи UML-діаграми класів, послідовності, стану та CRC картки.</p> <p>Курс також ставить на меті сформувати навички об'єктно-орієнтованого програмування на мові Java, включаючи створення класів, їх інкапсуляцію за допомогою модифікаторів доступу, декларування атрибутів і методів класів і їх екземплярів, використання інтерфейсів та абстрактних класів для наслідування змінних і методів.</p> <p>Окремо вивчаються основні принципи об'єктно-орієнтованого проектування з використанням породжувальних, поведінкових та структурних шаблонів проектування.</p>

<b>Сторінка курсу в MOODLE</b>	<a href="http://78.137.2.119:2929/course/view.php?id=539">http://78.137.2.119:2929/course/view.php?id=539</a>
<b>Мова викладання</b>	Українська
<b>Лектор курсу</b>	Ночевнов Дмитро Павлович, к.т.н., доцент Канали комунікації: СДН «Moodle»: повідомлення в чаті E-mail: dmytro.ndp@gmail.com
<b>Місце дисципліни в освітній програмі</b>	
<b>Освітня програма</b>	<a href="http://csbc.edu.ua/documents/otdel/oop_kb1.pdf">http://csbc.edu.ua/documents/otdel/oop_kb1.pdf</a>
<b>Перелік загальних компетентностей (ЗК)</b>	Z1. Здатність до абстрактного мислення, аналізу і синтезу. Z2. Здатність вчитися і оволодівати сучасними знаннями. Z3. Здатність застосовувати знання у практичних ситуаціях. Z4. Здатність спілкуватися державною мовою як усно, так і письмово. Z5. Здатність спілкуватися іноземною мовою. Z6. Навички міжособистісної взаємодії. Z7. Вміння виявляти, ставити та вирішувати проблеми. Z8. Здатність працювати в команді.
<b>Перелік спеціальних компетентностей (СК)</b>	P2. Здатність використовувати сучасні методи і мови програмування для розроблення алгоритмічного та програмного забезпечення. P3. Здатність створювати системне та прикладне програмне забезпечення комп'ютерних систем та мереж. P11. Здатність оформляти отримані робочі результати у вигляді презентацій, науково-технічних звітів. P13. Здатність вирішувати проблеми у галузі комп'ютерних та інформаційних технологій, визначати обмеження цих технологій.
<b>Перелік програмних результатів навчання</b>	N3. Знати новітні технології в галузі комп'ютерної інженерії. N4. Знати та розуміти вплив технічних рішень в суспільному, економічному, соціальному і екологічному контексті. N10. Вміти розробляти програмне забезпечення для вбудованих і розподілених застосувань, мобільних і гібридних систем, розраховувати, експлуатувати, типове для спеціальності обладнання. N11. Вміти здійснювати пошук інформації в різних джерелах для розв'язання задач комп'ютерної інженерії. N12. Вміти ефективно працювати як індивідуально, так і у складі команди. N13. Вміти ідентифікувати, класифікувати та описувати роботу комп'ютерних систем та їх компонентів. N14. Вміти поєднувати теорію і практику, а також приймати рішення та виробляти стратегію діяльності для вирішення

	<p>завдань спеціальності з урахуванням загальнолюдських цінностей, суспільних, державних та виробничих інтересів.</p> <p>N15. Вміти виконувати експериментальні дослідження за професійною тематикою.</p> <p>N16. Вміти оцінювати отримані результати та аргументовано захищати прийняті рішення.</p>
<b>Опис дисципліни</b>	
<b>Структура навантаження на студента</b>	<p>Загальна кількість годин – 90</p> <p>Кількість кредитів – 3</p> <p>Кількість лекційних годин – 15</p> <p>Кількість практичних занять – 30</p> <p>Кількість годин для самостійної роботи студентів – 45</p> <p>Форма підсумкового контролю – іспит</p>
<b>Методи навчання</b>	<p>Словесні (лекція, самостійна робота з джерелами інформації, науково-популярна розповідь);</p> <p>Наочні (презентаційні повідомлення)</p> <p>Практичні (лабораторні роботи);</p> <p>Інтерактивні методи (дистанційні консультації).</p>
<b>Зміст дисципліни</b>	
<b>Тема 1.</b> Базові поняття об'єктно-орієнтованого програмування. Моделювання з використанням UML	<p>Термінологія і фундаментальні концепції об'єктно-орієнтованого програмування. Атрибути і методи об'єктів. Зв'язки між класами. Unified Modeling Language (UML). UML-діаграми класів, послідовності, стану. Class-responsibility-collaboration картки.</p>
<b>Тема 2.</b> Об'єктно-орієнтований аналіз предметної області	<p>Визначення архітектури системи за допомогою об'єктно-орієнтованого аналізу. Виявлення варіантів використання. Моделювання класів і зв'язків між ними. Моделювання послідовності взаємодії між класами.</p>
<b>Тема 3.</b> Об'єктно-орієнтоване програмування на мові Java	<p>Послідовність запуску програми на мові Java. Типова структура класу в Java. Конструктори об'єктів. Декларування атрибутів і методів екземпляру об'єкту. Інкапсуляція класів за допомогою модифікаторів доступу. Використання інтерфейсів і абстрактних класів для наслідування. Декларування статичних методів і атрибутів класу.</p>
<b>Тема 4.</b> Основні принципи об'єктно-орієнтованого проектування	<p>Ознаки погано спроектованої програми. SOLID-принципи: Single Responsibility Principle, Open/Closed Principle, Liskov Substitution Principle, Interface Segregation Principle, Dependency Inversion Principle.</p>
<b>Тема 5.</b> Породжувальні шаблони проектування	<p>Переваги, недоліки, сфера і приклади застосування породжувальних шаблонів проектування "Одинак", "Прототип", "Фабричний метод", "Абстрактна фабрика", "Будівельник".</p>

<b>Тема 6.</b> Поведінкові шаблони проектування	Переваги, недоліки, сфера і приклади застосування шаблонів проектування "Ланцюжок обов'язків", "Команда", "Ітератор", "Стан", "Шаблонний метод".	
<b>Тема 7.</b> Структурні шаблони проектування	Переваги, недоліки, сфера і приклади застосування шаблонів проектування "Адаптер", "Фасад", "Декоратор", "Замісник", "Міст".	
<b>Політика дисципліни</b>		
<b>Політика відвідування</b>	Регулярне відвідування всіх видів занять, своєчасність виконання самостійної роботи. За об'єктивних причин (наприклад, хвороба, міжнародне стажування) навчання зорганізується в онлайн формі за погодженням із керівником курсу.	
<b>Політика щодо дедлайнів та перескладання</b>	Роботи, які здаються із порушенням термінів без поважних причин, оцінюються на нижчу оцінку.	
<b>Академічна доброчесність</b>	У випадку недотримання політики академічної доброчесності (плагиат, фабрикація, фальсифікація, списування, обман, хабарництво) передбачено повторне проходження оцінювання.	
<b>Система оцінювання</b>		
Поточний контроль здійснюється протягом семестру під час проведення лабораторно-практичних 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 іспиту, відповідно до графіку навчального процесу.		
<b>Накопичування рейтингових балів з навчальної дисципліни</b>		
<b>Види навчальної роботи</b>	<b>Мах кількість балів</b>	
Виконання лабораторних робіт	40	
Модульні контрольні роботи (2 к.р.)	20	
Індивідуальне завдання	10	
Завдання іспиту	30	
Разом	100	
<b>Шкала оцінювання</b>		
<b>ECTS</b>	<b>Бали</b>	<b>Зміст</b>
<b>A</b>	90-100	Бездоганна підготовка в широкому контексті
<b>B</b>	80-89	Повні знання, міцні вміння
<b>C</b>	70-79	Хороші знання та вміння
<b>D</b>	65-69	Задовільні знання, стереотипні вміння

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

### Список рекомендованих джерел

#### Основна

1. Буч, Гради, Максимчук, Роберт А., Энгл, Майкл У., Янг, Бобби Дж., Коналлен, Джим, Хьюстон, Келли А. Объектно-ориентированный анализ и проектирование с примерами приложений, 3-е изд. : Пер. с англ. — М. : ООО "И.Д. Вильямс", 2008. - 720 с.
2. Буч Г., Рамбо Д., Якобсон И. Язык UML. Руководство пользователя. 2-е изд.: Пер. с англ. Мухин Н. – М.: ДМК Пресс, 2006. – 496 с.: ил.
3. Мартин, Роберт С. - Быстрая разработка программ. Принципы, примеры, практика. - М. : Издательский дом "Вильямс", 2004. - 752 с.
4. Дэвид Флэнаган, Бенджамин Эванс. Java. Справочник разработчика. - Диалектика, 2019. - 592 с.

#### Додаткова

1. Вайсфельд Мэтт. Объектно-ориентированный подход. 5-е межд. изд. — СПб.: Питер, 2020. — 256 с.: ил.
2. Роберт Лигуори. Патрисия Лигуори. Java 8. Карманный справочник. - М.: ООО "И.Д.Вильямс", 2016. - 265 с.
3. Кетті Сьєрра, Берт Бейтс. Head First Java. Легкий для сприйняття довідник. - Видавництво : Фабула, 2022, - 720 с.