



ОБ'ЄКТНО-ОРІЄНТОВАНИЙ АНАЛІЗ ТА КОНСТРУЮВАННЯ ПРОГРАМНИХ СИСТЕМ

Робоча програма навчальної дисципліни (Силабус)

Реквізити навчальної дисципліни

Рівень вищої освіти	Перший (бакалаврський)
Галузь знань	12 Інформаційні технології
Спеціальність	121 Інженерія програмного забезпечення
Освітня програма	Інженерія програмного забезпечення інтелектуальних кібер-фізичних систем в енергетиці
Статус дисципліни	<u>Нормативна</u>
Форма навчання	<u>очна (денна)</u>
Рік підготовки, семестр	2 курс, <u>3 семестр</u>
Обсяг дисципліни	5,5 кред./165 год.(лекцій 36 год., практик. 36 год., СРС 93 год.)
Семестровий контроль/ контрольні заходи	Екзамен, МКР
Розклад занять	http://rozklad.kpi.ua/
Мова викладання	<u>Українська</u>
Інформація про керівника курсу / викладачів	Лектор: ст.викладач Сарибога Ганна Володимирівна, sarigana-eds@ill.kpi.ua, тел.: 0932358725 телеграм https://t.me/binarysvit Практичні : Сарибога Ганна Володимирівна
Розміщення курсу	<u>Платформа дистанційного навчання "Сікорський"</u> <u>Youtube</u>

Програма навчальної дисципліни

1. Опис навчальної дисципліни, її мета, предмет вивчення та результати навчання

Метою дисципліни «Об'єктно-орієнтований аналіз та конструювання програмних систем» є набуття знань та практичних навичок використання сучасних методів щодо об'єктно-орієнтованого підходу при розробці програмного забезпечення та принципів об'єктно-орієнтованого програмування мовою Python

Предметом дисципліни «Об'єктно-орієнтований аналіз та конструювання програмних систем» є серія підходів, інструментів і використання сучасних методів щодо розробки застосунків Інтернету речей та сенсорних мереж.

Основні завдання кредитного модуля.

Згідно з вимогами програми навчальної дисципліни студенти після засвоєння кредитного модуля мають продемонструвати наступні результати навчання.

Фахові компетентності:

- здатність ідентифікувати, класифікувати та формулювати вимоги до програмного забезпечення (ФК1);
- здатність брати участь у проектуванні програмного забезпечення, включаючи проведення моделювання (формальний опис) його структури, поведінки та процесів функціонування (ФК2);
- здатність розробляти архітектури, модулі та компоненти програмних систем (ФК 3),
- володіння знаннями про інформаційні моделі даних, здатність створювати програмне забезпечення для зберігання, видобування та опрацювання даних (ФК 7).

Програмні результати навчання:

- знати і застосовувати відповідні математичні поняття, методи доменного, системного і об'єктно-орієнтованого аналізу та математичного моделювання для розробки програмного забезпечення (ПРН5);
- вибирати вихідні дані для проектування, керуючись формальними методами опису вимог та моделювання (ПРН11);
- застосовувати на практиці ефективні підходи щодо проектування програмного забезпечення (ПРН12);
- застосовувати на практиці ефективні підходи щодо проектування програмного забезпечення (ПРН13).

2. Пререквізити та постреквізити дисципліни (місце в структурно-логічній схемі навчання за відповідною освітньою програмою)

Згідно з робочим навчальним планом навчальна дисципліна «Об'єктно-орієнтований аналіз та конструювання програмних систем» викладається студентам другого року підготовки першого (бакалаврського) рівня вищої освіти ступеня “бакалавр” спеціальності «Інженерія програмного забезпечення» освітньої програми «Інженерія програмного забезпечення інтелектуальних кіберфізичних систем в енергетиці» у третьому навчальному семестрі. Дисципліна «Об'єктно-орієнтований аналіз та конструювання програмних систем» забезпечується дисциплінами «Основи програмування», «Основи комп'ютерних мереж і систем» та забезпечує вивчення таких навчальних дисциплін як: «Архітектура системного програмного забезпечення».

3. Зміст навчальної дисципліни

Розділ 1. Об'єктно-орієнтований аналіз.

Тема 1.1. Об'єктно-орієнтований підхід до створення програмного забезпечення

Тема 1.2. Об'єктно-орієнтоване програмування в Python. Особливості

Тема 1.3. Можливості Python. Структури

Тема 1.4. Можливості Python. Модулі

Тема 1.5. Основи ООП. Класи, об'єкти в Python.

Тема 1.6. Принципи ООП в Python. Наслідування. Ітерація, абстракція, поліморфізм, інкапсуляція, конструктори, деструктори

Розділ 2. Статистична обробка та аналіз даних за допомогою Python

Тема 2.1. Обробка та коригування статистичних даних в csv файлах

Тема 2.2. Візуалізація даних з Pandas

Тема 2.3. Аналіз даних з matplotlib

Тема 2.4. Створення GUI з tkinter

Розділ 3. Методи створення програмного забезпечення(ПЗ)

Тема 3.1.Методологія розробки ПЗ

Тема 3.2 Вимоги до чистого коду

Розділ 4. Обробка та аналіз даних за допомогою алгоритмів машинного навчання

Тема 4.1.Алгоритми машинного навчання. Огляд можливостей

Тема 4.2. Алгоритми машинного навчання. k-найближчих сусідів

Тема 4.3. Алгоритми машинного навчання. Алгоритм класифікації CNN

Тема 4.4. Алгоритми розпізнавання мови

Тема 4.5. Алгоритми розпізнавання зображень

4. Навчальні матеріали та ресурси

Базова література

1. Костюченко А. О. Основи програмування мовою Python : навч. посіб. Чернігів : ФОП Баликіна С. М., 2020. 180 с.
2. Васильєв О. М. Програмування мовою Python. Тернопіль : Навчальна книга – Богдан,2019. 504 с.
3. Яковенко А. В. Основи програмування. Python. Частина 1 : підручник. Київ : КПІ ім. Ігоря Сікорського, 2018. 195 с.
4. Крєневич А. П. Python у прикладах і задачах. Частина 1. Структурне програмування : навч. посіб. Київ : ВПЦ "Київський Університет", 2017. 206 с.
5. Програмування числових методів мовою Python : підруч. / Анісімов А. В., Дорошенко А. Ю., Погорілий С. Д., Дорогий Я. Ю. Київ : Видавничо-поліграфічний центр "Київський університет", 2014. 640 с.

Додаткова література

1. Марченко О.О., Россада Т.В. Актуальні проблеми Data Mining: Навчальний посібник для студентів факультету комп'ютерних наук та кібернетики. – Київ. – 2017. – 150 с.
2. Ланде Д.В., Субач І.Ю., Бояринова Ю.Є. Основи теорії і практики інтелектуального аналізу даних у сфері кібербезпеки: навчальний посібник. – К.: ІСЗІ КПІ ім. Ігоря Сікорського», 2018. – 297 с.
3. Бахрушин В.Є. Методи аналізу даних : навчальний посібник для студентів. – Запоріжжя : КПУ, 2011. – 268 с.
4. Василенко О. А. Математично-статистичні методи аналізу у прикладних дослідженнях: навч. посіб. / О. А. Василенко, І. А. Сенча. – Одеса: ОНАЗ ім. О. С. Попова, 2011. – 166 с.

Навчальний контент

1. Методика опанування навчальної дисципліни (освітнього компонента)

Лекційні заняття

№ з/п	Назва теми лекції та перелік основних питань (перелік дидактичних засобів, посилання на інформаційні джерела)
1	2
1	Об'єктно-орієнтований підхід до створення програмного забезпечення <u>Основні питання:</u> елементи теорії об'єктно-орієнтованого програмування на Python, Наслідування, абстракція, інкапсуляція, реалізація конструкторів та деструкторів, поля класу, функції, специфікатори доступу.
2	Об'єктно-орієнтоване програмування в Python. Особливості <u>Основні питання:</u> методи та засоби аналізу програмного коду, проектування програми чи системи за допомогою об'єктно-орієнтованого підходу (DRY), включаючи візуальне моделювання. Візуалізація за допомогою мови UML (Unified Modeling Language) для моделювання систем на основі
3	Можливості Python. Структури <u>Основні питання:</u> Типи даних, оператори, конкатенація, управляючі структури, обробка винятків, функції
4	Можливості Python. Модулі <u>Основні питання:</u> 2. Модулі, пакети, кортежі. Регулярний вираз або RegEx
5	Основи ООП. <u>Основні питання:</u> Класи , об'єкти в Python.
6	Принципи ООП в Python. <u>Основні питання:</u> Наслідування. Ітерація, абстракція, поліморфізм, інкапсуляція, конструктори, деструктори
7	Обробка та коригування статистичних даних в csv файлах. <u>Основні питання:</u> Відкриття, зчитування, фільтрація, перевірка на помилки, внесення змін, створення власного набору даних
8	Візуалізація даних з Pandas <u>Основні питання:</u> інструмент для обробки та аналізу даних, операції фільтрації, сортування та агрегації.
9	Аналіз даних з matplotlib , seaborn <u>Основні питання:</u> Створення графіків, діаграм та інших візуальних представлень даних.
10	Створення GUI на Python за допомогою Tkinter <u>Основні питання:</u> Архітектура. Модулі Tkinter. Tkinter Life Preserve.
11	Методології розробки ПЗ <u>Основні питання:</u> Етапи створення проекту. Види та приклади методологій розробки програмного продукту та взаємодії із замовником: модель водоспаду (waterfall) , гнучка модель (agile) , модель великого вибуху (big-bang) ,RAD-модель, спіральна (SDLC) модель, інкрементна модель, XP та ін.
12	Вимоги до чистого коду. <u>Основні питання:</u> :Критерії та показники чистого коду. приклади. Створення якісного програмного продукту. Jira-інструмент для спільної роботи над проектом

1	2
13	Алгоритми машинного навчання. Огляд можливостей <u>Основні питання:</u> Що таке машинне навчання. Де і для чого використовується машинне навчання. Платформи для тестування алгоритмів машинного навчання. Види алгоритмів машинного навчання: k-найближчих сусідів, бінарне дерево, регресія, класифікація, глибоке навчання.
14	Алгоритми машинного навчання. k-найближчих сусідів <u>Основні питання:</u> Опис та можливості алгоритму. Застосування, тестування. CSV-файли. Набори даних. Тренування мережі. Прогнозування. Достовірність. Платформи Kaggle, Jupiter notebook
15	Алгоритми машинного навчання. Алгоритм Регресії <u>Основні питання:</u> Опис та можливості алгоритму. Застосування, тестування. CSV-файли. Набори даних. Тренування мережі. Прогнозування. Достовірність. Платформи Kaggle, Jupyter notebook
16	Алгоритми машинного навчання. Алгоритм класифікації CNN <u>Основні питання:</u> Опис та можливості алгоритму. Застосування, тестування. CSV-файли. Набори даних. Тренування мережі. Прогнозування. Достовірність. Платформи Kaggle, Jupyter notebook
17	Алгоритми розпізнавання мови <u>Основні питання:</u> Перетворення мови на текст та тексту на мову. Бібліотеки Google. Використання. Голосовий помічник
18	Алгоритми розпізнавання зображень <u>Основні питання:</u> Бібліотеки OpenCV, Tensorflow. Встановлення, використання, приклади, застосування.

Практичні заняття

Основні завдання циклу практичних занять полягають у набутті студентами практичних навичок у розробці програмного забезпечення на базі об'єктно-орієнтованого підходу та аналізу ІТ-продукту загалом.

№ з/п	Назва теми заняття
1	Оператори Python, конкатенація, введення\виведення, цикли
2	switch\case, оператори умови
3	Регулярні вирази, Python RegEx
4	Обробка винятків Python
5	Кортежі, списки Python
6	Функції Python, *args та **kwargs
7	Використання методу Python Set update() та словника в Python
8	Модулі, пакети
9	Метапрограмування з метакласами
10	Класи та об'єкти
11	Застосування принципів ООП
12	Обробка CSV файлів Python
13	Розробка інформаційно-пошукової системи з GUI
14	Пошук та тестування датасетів

1	2
15	Тренування нейромережі
16	Вступ до машинного навчання
17	Робота з платформою Kaggle
18	Робота з щоденником Jupyter

2. Самостійна робота студента

№ з/п	Назви тем і питань, що виносяться на самостійне опрацювання та посилання на навчальну літературу
Тема 1.2. Об'єктно-орієнтоване програмування в Python. Особливості https://www.javatpoint.com/python-tutorial	
1	Python SQL
2	Блокчейн на Python
3	Токенізатор на Python
4	Магічний метод Python
5	Декоратор Python
6	PySpark MLlib
7	Python Itertools
8	PyCharm
9	Python MySQL
Тема 4.5. Алгоритми розпізнавання зображень	
10	OpenCV в Python
Тема 4.1. Алгоритми машинного навчання. Огляд можливостей	
11	Розробка гри за допомогою Python Pygame https://www.javatpoint.com/library-in-python
Питання, що виносяться на самостійне опрацювання студентів, контролюються шляхом їх включення до переліку питань модульних контрольних робіт.	

Політика та контроль

1. Політика навчальної дисципліни (освітнього компонента)

Застосовуються стратегії активного і колективного навчання, які визначаються наступними методами і технологіями:

1) методи проблемного навчання (проблемний виклад, частково-пошуковий (евристична бесіда) і дослідницький метод);

2) особистісно-орієнтовані (розвиваючі) технології, засновані на активних формах і методах навчання («мозковий штурм», «аналіз ситуацій» дискусія, експрес-конференція);

3) інформаційно-комунікаційні технології, що забезпечують проблемно-дослідницький характер процесу навчання та активізацію самостійної роботи студентів (електронні презентації для лекційних занять, використання аудіо-, відео-підтримки навчальних занять);

4) лекційні та лабораторні заняття відносяться до аудиторних занять. Відвідування аудиторних занять є обов'язковим;

5) правила поведінки на заняттях: активність, підготовка коротких доповідей чи текстів, відключення телефонів, використання засобів зв'язку для пошуку інформації на гугл-диску викладача чи в інтернеті тощо;

6) правила захисту лабораторних робіт. На лабораторних заняттях студенти під керівництвом викладача вивчають методику експериментальних досліджень. На кожній лабораторній роботі студенти оформляють звіт у письмовому вигляді. До звіту заноситься перебіг досліду, його результати і даються пояснення отриманих результатів з урахуванням похибок експерименту;

7) індивідуальні завдання з дисципліни (реферати, розрахункові, графічні, тощо) видаються студентам в терміни, передбачені вищим навчальним закладом. Індивідуальні завдання виконуються студентом самостійно при консультуванні викладачем. Допускаються випадки виконання комплексної тематики кількома студентами;

8) правила призначення заохочувальних балів: своєчасне виконання та здача лабораторних, індивідуальних завдань, нестандартний підхід до вирішення певного завдання;

9) правила призначення штрафних балів: несвоєчасне виконання лабораторних та індивідуальних завдань, а також користування допоміжними засобами (наприклад, мобільний телефон, конспект лекцій) під час виконання контрольної роботи;

10) політика дедлайнів та перескладань: невчасно виконані та здані лабораторні роботи оцінюються нижчою оцінкою (-10-20% від загальної підсумкової оцінки);

11) політика щодо академічної доброчесності: письмові роботи можуть перевірятися на наявність плагіату і допускаються до захисту із коректними текстовими запозиченнями не більше 40%. Списування під час контрольних робіт та екзаменів заборонені;

12) інші вимоги, що не суперечать законодавству України та нормативним документам Університету:

– політика щодо відвідування: відвідування занять є обов'язковим компонентом оцінювання, за яке нараховуються бали. За об'єктивних причин (підтверджених документально) дозволяється перескладання пропущених тем курсу.;

– політика щодо виконання завдань: позитивно оцінюється відповідальність, старанність, креативність, фундаментальність.

2. Види контролю та рейтингова система оцінювання результатів навчання (PCO)

1. Оцінка з дисципліни виставляється за багатобальною системою, з подальшим перерахуванням у 4-бальну.

2. Максимальна кількість балів з дисципліни дорівнює 100.

3. Нарахування балів по окремих видах робіт:

Рейтинг студента з кредитного модуля складається з балів, що він отримав за:

- виконання практичних робіт;
- написання контрольної роботи (МКР).

3. Система рейтингових (вагових) балів та критерії оцінювання

1. Виконання практичних робіт

Оцінюються 18 робіт, передбачених робочою програмою.

Максимальний ваговий бал ЛР = 54.

Сума вагових балів практичних робіт:

№ з/п	Назва теми заняття	Максимальний ваговий бал
1	Оператори Python	3
2	Python RegEx	3
3	Список Python	3
4	Кортежі Python	3
5	Набори Python	3
6	Словник Python	3
7	Потік керування Python	3
8	Функції Python	3
9	ООП Python	3
10	Обробка винятків Python	3
11	Обробка файлів Python	3
12	Робота з даними в Python	3
13	Статистичний аналіз з використанням Python	3
14	Вступ до машинного навчання	3
15	Робота з платформою Kaggle	3
16	Робота з щоденником Jupiter	3
17	Робота з платформою Google collaboration	3
18	Вступ до Scikit-learn	3
	Разом	54

Оцінювання лабораторних робіт:

- якщо робота виконана невчасно знімається 10-30% від максимальної кількості балів (кількість процентів залежить від терміну запізнення);
- якщо робота виконана не самостійно та простежується не індивідуальне виконання то знімається 50% від максимальної кількості балів;
- якщо в програмі не витримані основні правила створення програмних продуктів (модульність, дружній інтерфейс, наявність коментарів та т.п.) знімається 5%.

2. Модульний контроль

На одному з лекційних занять проводиться модульна контрольна робота: Максимальний ваговий бал МКР = 6.

Оцінювання модульної контрольної роботи виконується наступним чином:

- якщо на всі питання дані повні та чітко аргументовані відповіді, контрольна виконана охайно, з дотримання основних правил, то виставляється 5- 6 балів;
- якщо методика виконання запропонованого завдання розроблена вірно, але допущені не принципові помилки у теоретичному описі або розрахунках, то виставляється 4 - 5 балів;
- від 3 до 5 балів нараховується, якщо методика виконання завдання розроблена в основному вірно, але допущені деякі з наступних помилок: помилки у представленні вихідних даних, не обгрунтовані теоретичні рішення, помилки у методиці розрахунків;
- нижче 3 балів нараховується, якщо завдання не виконане або допущені грубі помилки.

3. Екзамен

Екзамен відбувається у письмовій формі. Максимальна оцінка за екзамен складає $rEK = 40$ балів.

Умови позитивної проміжної атестації

Для отримання «зараховано» з першої проміжної атестації студент повинен мати не менше, ніж 12 балів (за умови, що за 8 тижнів згідно з календарним планом контрольних заходів студент повинен отримати 24 бали).

Для отримання «зараховано» з другої проміжної атестації студент повинен мати не менше, ніж 40 балів (за умови, що за 14 тижнів згідно з календарним планом контрольних заходів студент повинен отримати 60 балів).

Розрахунок шкали (R) рейтингу:

Сума вагових балів контрольних заходів протягом семестру складає:

$$R=54 +6+40 = 100 \text{ балів}$$

Таким чином, рейтингова шкала з кредитного модуля складає 100 балів.

Умови допуску до іспиту: зарахування всіх лабораторних робіт, а також стартовий рейтинг $r \geq 40$ балів.

Для отримання студентом відповідних оцінок (ECTS та традиційних) його рейтингова оцінка RD переводиться згідно таблиці.

Таблиця відповідності рейтингових балів оцінкам за університетською шкалою:

Кількість балів	Оцінка
100-95	Відмінно
94-85	Дуже добре
84-75	Добре
74-65	Задовільно
64-60	Достатньо
Менше 60	Незадовільно
Не виконані умови допуску	Не допущено

Додаткова інформація з дисципліни (освітнього компонента)

Методичні рекомендації

Для кращого засвоєння матеріалу дисципліни рекомендується використовувати на лекціях мультимедійні засоби навчання, які дозволяють інтенсифікувати навчальний процес, стимулювати розвиток мислення та уяви студентів, збільшувати обсяг навчального матеріалу для творчого засвоєння і використання його студентами, викликати зацікавленість та позитивне ставлення до навчання.

Методика побудована таким чином, що матеріал майже кожної лекції закріплюється виконанням завдання комп'ютерного практикуму. Завдання студенти отримують заздалегідь і на аудиторному занятті під керівництвом викладача виправляють помилки в разі їх наявності та відповідають на запитання щодо програмної реалізації та теоретичних засад роботи.

Якість самостійної роботи перевіряється на заняттях комп'ютерного практикуму.

Перелік питань, які виносяться на семестровий контроль:

1. Як перевернути рядок у Python
2. Як читати файл CSV на Python
3. Як запустити програму Python
4. Як приймати вхідні дані в Python

5. *Як перетворити список на рядок у Python*
6. *Як додати елемент у список*
7. *Як порівняти два списки в Python*
8. *Як перетворити int на рядок у Python*
9. *Як створити словник на Python*
10. *Як створити віртуальне середовище на Python*
11. *Як оголосити змінну в Python*
12. *Як встановити matplotlib на Python*
13. *Як встановити OpenCV на Python*
14. *Як друкувати в одному рядку в Python*
15. *Як читати файл JSON на Python*
16. *Як читати текстовий файл на Python*
17. *Як використовувати цикл for у Python*
18. *Як об'єднати два рядки в Python*
19. *Як підключити базу даних у Python*
20. *Як перетворити список на словник у Python*
21. *Як оголосити глобальну змінну в Python*
22. *Як перевернути число в Python*
23. *Що таке об'єкт у Python*
24. *Як очистити оболонку Python*
25. *Як створити DataFrames у Python*
26. *Як розробити гру на Python*
27. *Як встановити Tkinter на Python*
28. *Як побудувати графік на Python*
29. *Як надрукувати шаблон на Python*
30. *Як видалити елемент зі списку в Python*
31. *Як округлити число в Python*
32. *Як сортувати словник у Python*
33. *Як перетворити текст на мовлення в Python*
34. *Бульбашкове сортування в Python*
35. *Як отримати поточну дату в Python*
36. *argparse в Python*
37. *Модуль Python tqdm*
38. *Шифр Цезаря в Python*
39. *Токенізатор на Python*
40. *Як додати два списки в Python*
41. *Як перетворити байт на рядок у Python*
42. *Як перетворити список на набір*
43. *Python SQL*
44. *Блокчейн на Python*
45. *Токенізатор на Python*
46. *Магічний метод Python*
47. *Декоратор Python*
48. *PySpark MLlib*
49. *Python Itertools*
50. *PyCharm*
51. *Python MySQL*

Робочу програму навчальної дисципліни (силабус):

Складено *ст. викл. кафедри інженерії програмного забезпечення в енергетиці*

Ганна Сарибога

Ухвалено *кафедрою ІПЗЕ (протокол № 28 від 15.05.2023)*

Погоджено *НН ІАТЕ ім.Ігоря Сікорського (протокол № 9 від 26.05.2023)*