



Силабус освітнього компонента

Програма навчальної дисципліни



Базовий курс програмування Python

Шифр та назва спеціальності

121 – Інженерія програмного забезпечення
122 – Комп'ютерні науки

Інститут

ННІ Комп'ютерних наук та інформаційних технологій

Освітня програма

Інженерія програмного забезпечення
Комп'ютерні науки та інтелектуальні системи

Кафедра

Програмна інженерія та інтелектуальні технології управління (321)

Рівень освіти

Бакалавр

Тип дисципліни

Спеціальна (фахова), Вибіркова

Семестр

4

Мова викладання

Українська, англійська

Викладачі, розробники



Коваленко Світлана Миколаївна

Svitlana.Kovalenko@khpі.edu.ua

канд.техн.наук, доцент, доцент кафедри програмної інженерії та інтелектуальних технологій управління

Google Scholar:

<https://scholar.google.com/citations?user=jeD1w74AAAAJ&hl>

ORCID: <https://orcid.org/0000-0001-6770-6778>

Scopus: <https://www.scopus.com/authid/detail.uri?authorId=57212035934>

Web of Science: <https://www.webofscience.com/wos/author/record/F-8252-2017>

[Детальніше про викладача на сайті кафедри](#)

Загальна інформація

Анотація

Завданням дисципліни є набуття знань, умінь та навичок на рівні новітніх досягнень в області програмування мовою Python; вивчення базових концепцій, механізмів та технік процедурного та об'єктно-орієнтованого програмування мовою Python; розв'язування задач з використанням різних структур даних, умовних конструкцій, циклів, функцій, роботи з файлами, а також навички розробки та налагодження програмного коду з дотриманням кращих практик програмування, здобуття навичок проектування, розробки та тестування програмного забезпечення

Мета та цілі дисципліни

Формування у студентів теоретичних та практичних знань щодо синтаксису мови програмування Python, використання модулів її стандартної бібліотеки, а також застосування методів і техніки створення, відлагодження та тестування програм з використанням різних IDE

Формат занять

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

Компетентності

- 121-K01. Здатність до абстрактного мислення, аналізу та синтезу.
- 121-K02. Здатність застосовувати знання у практичних ситуаціях.
- 121-K05. Здатність вчитися і оволодівати сучасними знаннями.
- 121-K06. Здатність до пошуку, оброблення та аналізу інформації з різних джерел.
- 121-K13. Здатність ідентифікувати, класифікувати та формулювати вимоги до програмного забезпечення.
- 121-K15. Здатність розробляти архітектури, модулі та компоненти програмних систем.
- 121-K19. Володіння знаннями про інформаційні моделі даних, здатність створювати програмне забезпечення для зберігання, видобування та опрацювання даних.
- 121-K22. Здатність накопичувати, обробляти та систематизувати професійні знання щодо створення і супроводження програмного забезпечення та визнання важливості навчання протягом всього життя.
- 121-K23. Здатність реалізовувати фази та ітерації життєвого циклу програмних систем та інформаційних технологій на основі відповідних моделей і підходів розробки програмного забезпечення.
- 121-K25. Здатність обґрунтовано обирати та освоювати інструментарій з розробки та супроводження програмного забезпечення.
- 121-K26. Здатність до алгоритмічного та логічного мислення.
- 122-ЗК1. Здатність до абстрактного мислення, аналізу та синтезу.
- 122-ЗК2. Здатність застосовувати знання у практичних ситуаціях.
- 122-ЗК6. Здатність вчитися й оволодівати сучасними знаннями.
- 122-ЗК7. Здатність до пошуку, оброблення та аналізу інформації з різних джерел.
- 122-ЗК8. Здатність генерувати нові ідеї (креативність).
- 122-ЗК11. Здатність приймати обґрунтовані рішення.
- 122-СК8. Здатність проектувати та розробляти програмне забезпечення із застосуванням різних парадигм програмування: узагальненого, об'єктно-орієнтованого, функціонального, логічного, з відповідними моделями, методами й алгоритмами обчислень, структурами даних і механізмами управління.
- 122-СК9. Здатність реалізувати багаторівневу обчислювальну модель на основі архітектури клієнт-сервер, включаючи бази даних, знань і сховища даних, виконувати розподілену обробку великих наборів даних на кластерах стандартних серверів для забезпечення обчислювальних потреб користувачів, у тому числі на хмарних сервісах.
- 122-СК10. Здатність застосовувати методології, технології та інструментальні засоби для управління процесами життєвого циклу інформаційних і програмних систем, продуктів і сервісів інформаційних технологій відповідно до вимог замовника.
- 122-СК12. Здатність забезпечити організацію обчислювальних процесів в інформаційних системах різного призначення з урахуванням архітектури, конфігурування, показників результативності функціонування операційних систем і системного програмного забезпечення.

Результати навчання

- 121-ПР01. Аналізувати, цілеспрямовано шукати і вибирати необхідні для вирішення професійних завдань інформаційно-довідникові ресурси і знання з урахуванням сучасних досягнень науки і техніки.
- 121-ПР02. Знати кодекс професійної етики, розуміти соціальну значимість та культурні аспекти інженерії програмного забезпечення і дотримуватись їх в професійній діяльності.
- 121-ПР08. Вміти розробляти людино-машинний інтерфейс.
- 121-ПР12. Застосовувати на практиці ефективні підходи щодо проектування програмного забезпечення.
- 121-ПР13. Знати і застосовувати методи розробки алгоритмів, конструювання програмного забезпечення та структур даних і знань.
- 121-ПР15. Мотивовано обирати мови програмування та технології розробки для розв'язання завдань створення і супроводження програмного забезпечення.
- 121-ПР18. Знати та вміти застосовувати інформаційні технології обробки, зберігання та передачі даних.

121-ПР23. Вміти документувати та презентувати результати розробки програмного забезпечення.

122-ПР1. Застосовувати знання основних форм і законів абстрактно-логічного мислення, основ методології наукового пізнання, форм і методів вилучення, аналізу, обробки та синтезу інформації в предметній області комп'ютерних наук.

122-ПР4. Використовувати методи обчислювального інтелекту, машинного навчання, нейромережевої та нечіткої обробки даних, генетичного та еволюційного програмування для розв'язання задач розпізнавання, прогнозування, класифікації, ідентифікації об'єктів керування тощо.

122-ПР10. Використовувати інструментальні засоби розробки клієнт-серверних застосувань, проектувати концептуальні, логічні та фізичні моделі баз даних, розробляти та оптимізувати запити до них, створювати розподілені бази даних, сховища та вітрини даних, бази знань, у тому числі на хмарних сервісах, із застосуванням мов веб-програмування.

122-ПР11. Володіти навичками управління життєвим циклом програмного забезпечення, продуктів і сервісів інформаційних технологій відповідно до вимог і обмежень замовника, вміти розробляти проєктну документацію (техніко-економічне обґрунтування, технічне завдання, бізнес-план, угоду, договір, контракт).

122-ПР20. Розробляти архітектуру програмних систем та їх окремих компонент при побудові інтелектуальних систем управління у різних галузях, а також управляти процесами життєвого циклу програмного забезпечення інтелектуальних систем управління.

Обсяг дисципліни

Загальний обсяг дисципліни 150 год. (5 кредитів ECTS): лекції – 32 год., лабораторні роботи – 32 год., самостійна робота – 86 год.

Передумови вивчення дисципліни (пререквізити)

Вивчення цієї дисципліни безпосередньо спирається на: "Об'єктно-орієнтоване програмування", "Основи програмування", "Алгоритмізація та програмування".

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 заліку, відповідно до графіку навчального процесу (FAS).

Програма навчальної дисципліни

Теми лекційних занять

Тема 1. Введення до Python. Основи синтаксису Python, Дзен Python

Компілятори та інтерпретатори. Python як інтерпретована мова програмування. Переваги та недоліки Python. Межі застосування. Історія створення. Версії. Встановлення. IDE. Дзен Python. Ключові слова. Регістрозалежність та правила побудови інтерпретаторів. PEP8. Вирази та оператори. Ознака закінчення оператора. Відступи як ознака блоку коду. Функція print(). Python як мова з сильною динамічною типізацією. Основні типи даних в Python. Поняття змінних та незмінних типів даних. Зміни як вказівники. Типи операторів в Python

Тема 2. Оператори порівняння, логічні оператори, умовні конструкції

Компаратори. Особливості використання операторів порівняння в Python. Логічні оператори та їх особливості. Пріоритет операцій. if оператор, складові elif та else. Умовний вираз. Моржовий оператор та його використання. Оператор match.

Тема 3. Цикли в Python.

Цикл while. Нескінченний цикл та варіанти його використання. Цикл for та його особливості. Функція range(). Оператори break та continue. Гілка else.

Тема 4. Списки в Python, оператори, методи та вбудовані функції для роботи з ними. Дослідження продуктивності коду

Поняття списків. Основні властивості списків Python. Основні операції зі списками: додавання, видалення, індексація елементів. Зрізи: поняття, варіанти використання. Функції `in` та `not in`. Спискові включення. Поняття функції та метода в Python. Вбудовані функції та методи для роботи зі списками. Дослідження продуктивності коду.

Тема 5. Функції користувача та робота з ними

Типи функцій в Python. Синтаксис користувацької функції. Doc strings. Типи параметрів функцій. Область видимості параметра. Ключове слово `global`. Рекурсія. Вкладені функції.

Тема 6. Рядки в Python.

Збіжність та відмінність рядків та списків. Способи створення рядкових літералів. Оператори, методи та вбудовані функції для роботи з рядками. Міні мова форматування рядків в Python. `f`-рядки.

Тема 7. Кортежі, словники та множини. Спискові, словникові та множинні включення, вирази-генератори

Властивості кортежів, їх відмінність від списків, способи створення кортежів. Оператори, методи та вбудовані функції для роботи з кортежами. Операція розпакування. Оператор `enumerate()`. Словники. Властивості словників. Обмеження на ключі словників. Елементи словників та доступ до них. Методи словників. Властивості множин. Обмеження на елементи множини. Операції з елементами множин. Математичні операції з множинами. `Frozenset` та його властивості. Основи роботи з списковими, словниковими та множинними включеннями та виразами-генераторами.

Тема 8. Опрацювання виключень та оператор `assert`.

Виключення та синтаксичні помилки. Вбудовані виключення. Опряцювання виключень за допомогою `try` та `except` в Python. Гілки `else` та `finally`. Користувацький виклик виключень. Оператор `assert` та варіанти його використання.

Тема 9. Лямбда функції, функції `map()`, `filter()`, `reduce()` та `zip()`.

Поняття, створення та використання лямбда-функції. Приклади використання `map()` та `filter()`. Використання лямбда функцій у комбінації з `map()` та `filter()`. Варіанти використання `reduce()` та `zip()`.

Тема 10. Об'єктно-орієнтоване програмування в Python.

Поняття класу та екземпляру класу. Ключове слово `class`. Method Resolution Order. Змінні екземпляру класу. Ключове слово `self`. Змінні класу. Методи класу, екземпляру класу та статичні. Наслідування в Python. Поняття батьківського та дочірнього класу. Типи наслідування. Функція `super()`. Магічні методи. `@property` decorator.

Теми практичних занять

Практичні заняття в рамках дисципліни не передбачені.

Теми лабораторних робіт

Тема 1. Початок роботи з Python та дослідження IDE

Тема 2. Логічні вирази та `if` оператор

Тема 3. Використання циклів в Python

Тема 4. Робота зі списками Python

Тема 5. Використання функцій користувача та порівняння продуктивності коду

Тема 6. Робота з рядками у Python

Тема 7. Використання словників Python

Тема 8. Створення та використання класів

Самостійна робота

Студентам рекомендуються додаткові матеріали (відео, статті) для самостійного вивчення та опрацювання.

Література та навчальні матеріали

Основна література

1. Luciano Ramalho. Fluent Python, O'Reilly Media, 2022, 980p
2. Eric Matthes Python Crash Course, 2nd Edition, No Starch Press, 2019, 548p.
3. Richard L. Halterman. Fundamentals of Python Programming, 2019, 669p.
4. Dusty Phillips, Steven F. Lott Python Object-Oriented Programming, 4th edition, Packt, 2019, 714p
5. J. Hunt: A Beginners Guide to Python 3 Programming. Springer, 2019. 433 p.

Додаткова література

1. Al Sweigart. Automate the Boring Stuff with Python, 2nd edition: Practical Programming for Total Beginners, No Starch Press, 2020, 901p.
2. Al Sweigart. Python Programming Exercises, Gently Explained. 2022, 160p.
3. Allen B. Downey. Think Python 2e [Електронний ресурс] – Режим доступу: <https://greenteapress.com/wp/think-python-2e/>
4. B. Stephenson: The Python Workbook: A Brief Introduction with Exercises and Solutions, 2 nd ed. // Springer, 2019
5. John Canning, Alan Broder, Robert Lafore. Data Structures & Algorithms in Python, Pearson Education, Inc., 2023, 1050p.

Система оцінювання

Критерії оцінювання успішності студента та розподіл балів

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

- 8 лабораторних робіт (по 6%);
- 2 контрольні роботи (по 6%);

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

08.06.2023

Завідувач кафедри
Ігор ГАМАЮН

08.06.2023

Гаранти ОП

Андрій КОПІ
Юлія ЛІТВИНОВА

