

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Львівський національний університет імені Івана Франка
Факультет електроніки та комп'ютерних технологій
Кафедра оптоелектроніки та інформаційних технологій

Затверджено

На засіданні
кафедри оптоелектроніки та
інформаційних технологій
факультету електроніки та комп'ютерних
технологій
Львівського національного університету
імені Івана Франка
(протокол № від 2022 р.)

Завідувач кафедри:

_____ Олег КУШНІР

Силабус з навчальної дисципліни
“ Професійний Python”,
що викладається в межах ОПП
“ Інформаційні системи та технології ”
першого (бакалаврського) рівня вищої освіти для здобувачів з
спеціальності 126 – Інформаційні системи та технології

Львів 2022 р.

Назва дисципліни	Професійний Python
Адреса викладання дисципліни	Корпус факультету електроніки та комп'ютерних технологій, Львівський національний університет імені Івана Франка, вул. Драгоманова 50, м. Львів, 79005, вул. Ген. Тарнавського 107, м. Львів, 79011
Факультет та кафедра, за якою закріплена дисципліна	Факультет електроніки та комп'ютерних технологій Кафедра оптоелектроніки та інформаційних технологій
Галузь знань, шифр та назва спеціальності	12 – інформаційні технології 126 – Інформаційні системи та технології
Викладачі дисципліни	Свебеба С.А., доктор фізико-математичних наук, старший науковий співробітник, професор кафедри оптоелектроніки та інформаційних технологій
Контактна інформація викладачів	serhiy.sveleba@lnu.edu.ua, https://electronics.lnu.edu.ua/employee/sveleba-serhij-andrijovych
Консультації з питань навчання по дисципліні відбуваються	Консультації в день проведення лекцій/лабораторних занять (за попередньою домовленістю): ауд. 213, корпус факультету електроніки та комп'ютерних технологій, вул. Ген. Тарнавського 107, м. Львів. Також можливі он-лайн консультації через MS Teams. Для погодження часу он-лайн консультацій слід писати на електронну пошту викладача.
Сторінка курсу	https://moodle.elct.lnu.edu.ua/course/view.php?id=51
Коротка анотація дисципліни	Дисципліна “Професійний Python” є вибірковою навчальною дисципліною з спеціальності 126 – інформаційні системи та технології для освітньої програми “Інформаційні системи та технології”, яка викладається в 6-му семестрі в обсязі 3.5 кредитів (за Європейською Кредитно-Трансферною Системою ECTS).
Інформація про дисципліну	В курсі розглядаються базові поняття структурного програмування на мові Python 3.X. Формування у студентів базових понять структурного програмування на Python 3.X, вміння застосовувати їх на практиці.
Мета та цілі дисципліни	<i>Метою</i> є оволодіння студентами основних принципів структурного програмування, принципами побудови функцій користувача, з використанням локальних і глобальних змінних; організацію рекурсивних функцій користувача, освоїти методики визначення та практичного застосування рекурсивних функцій у програмах; основні положення роботи з бінарними та текстовими файлами, модульний принцип організації програми; основи побудови сховищ даних. <i>Цілі:</i> формування у студентів базових понять структурного програмування на Python 3.X, вміння застосовувати їх на практиці.
Література для вивчення дисципліни	1. Програмування числових методів мовою PYTHON / А. Ю. Дорошенко [та ін.]; за ред. А. В. Анісімова. – ВПЦ "Київський університет", 2013. – 464 с. 2. Бизли Дэвид М. Язык программирования Python : [пер. с англ.] / Дэвид М. Бизли. – К. : ДиаСофт, 2000. – 336 с

	<p>3. Програмування числових методів мовою Python : підруч. / А. В. Анісімов, А. Ю. Дорошенко, С. Д. Погорілий, Я. Ю. Дорогий ; за ред. А. В. Анісімова. – К. : Видавничо-поліграфічний центр "Київський університет", 2014. – 640 с.</p> <p>4. Сучасні методи аналітичного і числового програмування в задачах теоретичної фізики: метод. рекомендації / укл.: Ю.О. Сеті. – Чернівці : Чернівецький нац. ун-т, 2021. – 69 с.</p> <p>5. Методичні вказівки до виконання лабораторних робіт з навчальної дисципліни “Технології розробки алгоритмів” для студентів денної та заочної форми навчання спеціальностей 6.050102 та 123 “Комп’ютерна інженерія”, 6.170103 та 125 “Кібербезпека” / уклад. Гермак В.С.; Кропивницький: ЦНТУ – 2018.– 81 с.</p> <p>6. Методичні вказівки до виконання лабораторних робіт з дисципліни “Інженерія прикладних інтелектуально-орієнтованих програмних продуктів” для студентів спеціальностей 121 “Інженерія програмного забезпечення” та 122 “Комп’ютерні науки та інформаційні технології” (всіх форм навчання) / В.М. Льовкін. – Запоріжжя : ЗНТУ, 2016. – 80 с.</p> <p>7. Лутц М. Изучаем Python. 4-е издание. – Пер. с англ. – СПб.: СимволПлюс, 2011. – 1280 с.</p> <p>8. Лутц М. Программирование на Python, том 1, 4-е издание. – Пер. с англ. – СПб.: Символ-Плюс, 2011. – 992 с. , ил. [гл. 7–10]</p> <p>9. Downey, A.B. Think Python [Текст] / Allen B. Downey. – O’Reilly, 2012. – 300 p.</p> <p>10. Phillips, D. Python 3 Object Oriented Programming [Текст] / Dusty Phillips. – Birmingham : Packt Publishing, 2010. – 404 p.</p>
Обсяг курсу	128 години аудиторних занять. З них 32 години лекцій, 32 години лабораторних робіт та 64 година самостійної роботи
Очікувані результати навчання	<p>Після завершення цього курсу студент буде:</p> <p><i>Знати</i> основні поняття структурного програмування; базові вбудовані типи об’єктів структурного програмування (числа, рядки, списки, словники, кортежі, файли, множини) і синтаксичні конструкції використання цих об’єктів у вигляді літералів (виразів, які генерують ці об’єкти); елементи структурного програмування; принципи побудови функцій; модульний принцип організації програми;основи побудови сховищ даних</p> <p><i>Вміти</i> використовувати базові поняття структурного програмування для розв’язання різних задач; створювати і обробляти структури, масиви структур, найпростішу обробку файлів, створювати і використовувати функції.</p> <p>Після вивчення даного курсу «Професійний Python» здобувачі набудуть таких Загальних та Фахових компетентностей та Програмних результатів навчання:</p> <p>K01. Здатність до абстрактного мислення, аналізу та синтезу.</p> <p>K02. Здатність застосовувати знання у практичних ситуаціях.</p> <p>K03. Здатність спілкуватися державною мовою як усно, так і письмово.</p> <p>K04. Здатність спілкуватися іноземною мовою мовою як усно, так і письмово.</p> <p>K14. Здатність брати участь у проектуванні програмного забезпечення, включаючи проведення моделювання (формальний опис) його структури, поведінки та процесів функціонування.</p>

	<p>K15. Здатність розробляти архітектури, модулі та компоненти програмних систем.</p> <p>K20. Здатність застосовувати фундаментальні і міждисциплінарні знання для успішного розв'язання завдань інженерії програмного забезпечення.</p> <p>K26. Здатність до алгоритмічного та логічного мислення</p> <p>ПР05: Знати і застосовувати відповідні математичні поняття, методи доменного, системного і об'єктно-орієнтованого аналізу та математичного моделювання для розробки програмного забезпечення.</p> <p>ПР23: Вміти документувати та презентувати результати розробки програмного забезпечення.</p>
Ключові слова	Цикли, масиви, вбудовані функції, функції користувача, класи, атрибути класу, об'єкти класу
Формат курсу	Очний
Теми	<p>Тема 1. Вступ в програмування на Python</p> <p>Тема 2. Основні стандартні модулі Python</p> <p>Тема 3. Елементи функціонального програмування.</p> <p>Тема 4. Елементи ООП.</p> <p>Тема 5. Чисельні алгоритми.</p> <p>Тема 6. Обробка текстів</p> <p>Тема 7. Робота з даними в різних форматах.</p> <p>Тема 8. Розробка WEB додатку.</p> <p>Тема 9. Мережеві додатки на Python.</p> <p>Тема 10. Робота з базою даних</p> <p>Тема 11. Багатопотокові обчислення.</p> <p>Тема 12. Створення додатків з GUI.</p> <p>Тема 13. Інтеграція Python з іншими мовами програмування C</p> <p>Тема 14. Пристрій інтерпретатора мови Python</p> <p>Тема 15. Спеціальні методи класів у Python</p> <p>Тема 16. Методи файлових об'єктів</p>
Підсумковий контроль, форма	Залік в кінці семестру
Пререквізити	<p>Для вивчення даного курсу студентам потрібні базові знання з курсів:</p> <ul style="list-style-type: none"> - дискретна математика; - теорія алгоритмів; - алгоритмізація і програмування.
Навчальні методи та техніки, які будуть використовуватися під час викладання курсу	Інформаційні методи (лекція, бесіда, ілюстрація, демонстрація); дедуктивні методи на основі узагальнень; евристичні методи (проблемна лекція); інтерактивні методи).
Необхідне обладнання	Комп'ютер із необхідним програмним забезпеченням, доступ до Internet мережі.
Критерії оцінювання (окремо для кожного виду навчальної діяль-	<p>Оцінювання проводиться упродовж семестру за 100-бальною шкалою. Бали нараховуються за такими видами робіт з наступним співвідношенням:</p> <ul style="list-style-type: none"> • лабораторні роботи: 44% семестрової оцінки; максимальна кількість балів 44.

ності)	<ul style="list-style-type: none"> • контрольні заміри (3 модулі): 56% семестрової оцінки; максимальна кількість балів 56. <p>Загалом упродовж семестру 100 балів.</p> <p>Академічна доброчесність: Очікується, що роботи студентів будуть оригінальними дослідженнями чи міркуваннями. Списування та втручання в роботу інших студентів становлять, але не обмежують, приклади можливої академічної недоброчесності. Виявлення ознак академічної недоброчесності в написанні завдань є підставою для її незарахування викладачем, незалежно від масштабів плагіату чи обману.</p> <p>Відвідання занять є важливою складовою навчання. Очікується, що всі студенти відвідають усі лекції та лабораторні зайняття курсу. Студенти повинні інформувати викладача про неможливість відвідати заняття. У будь-якому випадку студенти зобов'язані дотримуватися термінів визначених для виконання всіх видів робіт, передбачених курсом.</p> <p>Література. Уся література, яку студенти не зможуть знайти самостійно, буде надана викладачем виключно в освітніх цілях без права її передачі третім особам. Студенти заохочуються до використання також й іншої літератури та джерел, яких немає серед рекомендованих.</p> <p>Політика виставлення балів. Враховуються бали, набрані при поточному контролі та бали підсумкового тестування. При цьому обов'язково враховуються присутність на заняттях та активність студента під час практичного заняття; недопустимість пропусків та запізнь на заняття; користування мобільним телефоном, планшетом чи іншими мобільними пристроями під час заняття в цілях не пов'язаних з навчанням; списування та плагіат; несвоєчасне виконання поставленого завдання і т. ін. Жодні форми порушення академічної доброчесності не толеруються.</p>
Питання до заліку чи екзамену.	<p>Типи даних, які розпізнає <i>Python</i>.</p> <p>Опишіть три варіанти використання функції range. Яку функцію замість range рекомендується застосовувати у разі великого розміру діапазону? Чому це може бути важливо?</p> <p>Чи можуть списки містити інші списки як елементи?</p> <p>Яким чином може завершитися виконання циклу?</p> <p>Яким чином кодуються логічні значення в мові <i>Python</i>? Чи має <i>Python</i> окремий логічний тип?</p> <p>Які функції можна використовувати для введення й виведення даних? Оптимізації циклів.</p> <p>Які методи для виклику об'єктів-списків? Їх призначення.</p> <p>Основні характеристики функціонального програмування. Як підтримується FP мовою <i>Python</i>?</p> <p>Яким чином передаються параметри у функції <i>Python</i>?</p> <p>Для чого документувати функції?</p> <p>Якими способами можна отримати доступ до документації з мови <i>Python</i>?</p> <p>Які є можливості генерування списків у мові <i>Python</i>?</p> <p>Яким чином визначається кортеж, що містить один елемент?</p> <p>Характеристика структуру даних "словник".</p> <p>Поняття "модуль" і "пакет"? Який між ними зв'язок?</p> <p>Варіанти використання інструкції <code>import</code>?</p> <p>Чому не варто використовувати інструкцію <code>from <модуль> import *</code>?</p> <p>Яким чином здійснюється пошук модулів інтерпретатором?</p> <p>Як можна зробити модулі видимими для інтерпретатора?</p> <p>Яким чином можна отримати перелік імен, які визначено в певному моду-</p>

	<p>лі?</p> <p>Опишіть можливості Python із форматованого виведення.</p> <p>Чим функція <code>str()</code> відрізняється від функції <code>repr()</code>?</p> <p>Функції для роботи з файлами і методи файлових об'єктів.</p> <p>Виняткові ситуації і яким чином здійснюється їх обробка у Python?</p> <p>Гілка <code>finally</code> в інструкції <code>try</code>? Чи можна її поєднувати з гілками <code>except</code>?</p> <p>Які два класи виняткових ситуацій наявні в Python? Який із них рекомендується використовувати у програмах?</p> <p>Області види мостів імен у Python.</p> <p>Які концепції ООП підтримує об'єктна модель Python?</p> <p>Які обмеження є в моделі ООП, реалізованій у Python?</p> <p>Механізми спадкування, реалізовані в мові Python.</p> <p>Оператори, що застосовуються до всіх об'єктів послідовностей.</p> <p>Оператори, що застосовуються тільки до списків, кортежів та словників.</p> <p>Вбудовані функції мови Python.</p> <p>Інструкції для організації циклів наявні в Python?</p> <p>У якому разі виконується гілка <code>else</code> для циклу <code>while</code>?</p> <p>Поясніть переваги використання механізму просторів імен.</p> <p>Як Python визначає логічне значення для об'єкта?</p> <p>Перерахуйте основні логічні операції.</p> <p>Який порядок обчислення виразів у операторі <code>and</code>?</p> <p>Чи можна порівнювати об'єкти довільних типів?</p> <p>Перерахуйте числові типи, які містить Python, і опишіть область використання кожного з них.</p> <p>Наведіть та опишіть арифметичні операції.</p> <p>Для чого можна використовувати бітові операції?</p> <p>Перерахуйте незмінювані послідовності, які є в мові Python.</p> <p>Які об'єкти можна використовувати як ключі до словника?</p> <p>Перерахуйте й опишіть операції над змінюваними послідовностями. Які виняткові ситуації можуть виникати при їх використанні?</p> <p>Опишіть викликувані об'єкти мови Python.</p> <p>Поясніть переваги використання спеціальних методів класів у мові Python.</p> <p>Дайте огляд бібліотечних модулів мови Python.</p> <p>Для чого може використовуватись переведення об'єктів у послідовну форму (серіалізація)?</p> <p>Перерахуйте основні пакети для створення графічного інтерфейсу користувача, що застосовуються з Python.</p> <p>Чому пакети графічного інтерфейсу Python – це в основному обгортки для бібліотек, що написані іншими мовами програмування?</p> <p>Перерахуйте основні віджети пакета Tkinter.</p> <p>Опишіть особливості обробки подій у пакеті Tkinter.</p> <p>Поясніть концепцію слотів і сигналів, яка використовується в бібліотеці PyQt4.</p> <p>Перерахуйте основні віджети бібліотеки Qt.</p> <p>Як буде відображатися віджет, для якого при створенні не вказано батька?</p> <p>Перерахуйте схеми розташування, що використовуються в пакеті PyQt4, та поясніть особливості їхнього застосування. Екземпляр якого класу має обов'язково бути в програмі, що використовує бібліотеку PyQt4.</p> <p>Перерахуйте й опишіть основні операції з рядками, наявні в мові Python.</p> <p>Яким чином можна змінювати рядки, якщо вони є незмінюваним типом даних?</p>
--	---

	<p>Як указати кодування, що використовується для тексту Python- програми?</p> <p>Яким чином із рядком можна працювати як із файлом?</p> <p>Що таке регулярні вирази?</p> <p>Перерахуйте основні символи загального призначення, що використовуються в регулярних виразах, та охарактеризуйте їх.</p> <p>Опишіть основні методи об'єкта-шаблону.</p> <p>Яким чином можна перекодувати деякий файл із кодуванням koï- 8u у файл із кодуванням sr866?</p> <p>Яким чином можна отримувати дані з html-сторінок? Формалізуйте необхідні для цього кроки.</p> <p>Які особливості формату CSV? У чому полягають його переваги й вади?</p> <p>Опишіть призначення та функції пакета email.</p> <p>Коротко опишіть призначення модулів, з яких складається пакет email.</p> <p>Охарактеризуйте переваги й недоліки використання мови XML</p> <p>Які існують підходи до аналізу XML-документів? Назвіть їхні переваги та вади.</p>
Опитування	<p>Анкету-оцінку з метою оцінювання якості курсу буде надано по завершенню курсу.</p>

Схема курсу “ Професійний Python ”
для студентів спеціальності 121 – Інженерія програмного забезпечення

Тиж.	Тема, план, короткі тези	Форма діяльності (заняття)* *лекція, самостійна, дискусія, групова робота)	Література.** * Ресурси в інтернеті	Завдання, год	Термін виконання
1	Вступ в програмування на Python. Програма на Python Основні алгоритмічні конструкції Послідовність операторів Оператор умови і вибору Цикли Функції Винятки Вбудовані типи даних Послідовності Взяття елемента за індексом і зрізи Вирази Імена Стиль програмування	Лекція	[1], [2], [3],[8], [9],	2	кінець поточного тижня
1	Лаб.1 Алгоритми циклічної структури (інструкція while)	Лабораторна робота		2	кінець поточного тижня
2	Основні стандартні модулі Python. Поняття модуля Модулі в Python Вбудовані функції Функції перетворення типів і класи Числові і рядкові функції Функції даних Функції визначення властивостей Функції для доступу до внутрішніх структур Функції компіляції та виконання Функції вводу-виводу	Лекція	[1], [3], [10],	2	кінець поточного тижня
2	Лаб.2. Алгоритми циклічної структури (інструкція for)	Лабораторна робота		2	кінець поточного тижня
3	Основні стандартні модулі Python. Функції для роботи з атрибутами функції- "прикрашувачі" методів класу Огляд стандартної бібліотеки Сервіси періоду виконання Модуль сорту Підтримка циклу розробки Взаємодія з операційною системою	Лекція	[1], [2], [10],	2	кінець поточного тижня
3	Лаб.3. Списки: одновимірні та двовимірні масиви. Генерація випадкових чисел	Лабораторна робота		2	кінець поточного тижня
4	Елементи функціонального програмування. Що таке функціональне програмування? Функціональна програма Функція: визначення і виклик Рекурсія Функції як параметри і результат Обробка послідовностей спискові включення Ітератори Власний ітератор Прості генератори Генераторний вираз каррінг	Лекція	[1], [2], [10],	2	кінець поточного тижня
4	Лаб.4 Рядки символів. Множини	Лабораторна робота		2	кінець поточного тижня
5	Елементи ООП. Основні поняття Абстракція і декомпозиція Об'єкти Типи і класи Визначення класу Інкапсуляція Доступ до властивостей Приховування даних Поліморфізм Імітація типів Відносини між класами Спадкування Множинне успадкування Порядок дозволу методів Агрегація Контейнери Ітератори Асоціація Статичний метод Метод класу метакласом мультиметод Стійкі об'єкти	Лекція	[1], [2], [3],	2	кінець поточного тижня

5	Лаб.5. Функції користувача. Наближене обчислення функцій	Лабораторна робота		2	кінець поточного тижня
6	Чисельні алгоритми. Матричні обчислення. Модуль Numeric Створення масиву Методи масивів Зрізи Універсальні функції	Лекція	[1], [2], [3],	2	кінець поточного тижня
6	Модуль 1.	Модуль		2	кінець поточного тижня
7	Чисельні алгоритми. Функції модуля Numeric Функції для роботи з масивами Звід функцій модуля Numeric Модуль LinearAlgebra Модуль RandomArray	Лекція	[1], [2], [3], [5],	2	кінець поточного тижня
7	Лаб.6. Рекурсивні функції. Документування коду.	Лабораторна робота		2	кінець поточного тижня
8	Обробка текстів. Регулярні вирази. Unicode. Рядки Кодування Python-програми Строкові літери Операції над рядками Операція форматування Модуль string Методи рядків Рекомендації по ефективності Модуль StringIO Модуль difflib Регулярні вирази Синтаксис регулярного виразу Методи об'єкта-шаблону Приклади шаблонів Налаштування регулярних виразів Приклади застосування регулярного виразу Обробка балки Аналіз записи числа Множинна заміна Робота з декількома файлами Робота з Unicode значень; генератори списків.	Лекція	[1], [2], [6], [10],	2	кінець поточного тижня
8	Лаб. 7. Словник. Кортеж	Лабораторна робота		2	кінець поточного тижня
9	Робота з даними в різних форматах. Формат CSV Пакет email Розбір повідомлення. Клас Message Формування повідомлення Розбір поля заголовка Мова XML Формування XML-документа Аналіз XML-документа Простору імен	Лекція	[1], [2], [3], [10],	2	Кінець поточного тижня
9	Лаб.8 Файли даних. Модульний принцип організації програми.	Лабораторна робота		2	кінець поточного тижня
10	Розробка WEB додатку. CGI-сценарії Модуль cgi Середовища розробки Zope і його об'єктна модель	Лекція	[1], [2], [3],	2	кінець поточного тижня
10	Модуль 2	Модуль		2	кінець поточного тижня
11	Мережеві додатки на Python. Робота з сонетами Модуль smtplib Модуль poplib Модулі для клієнта WWW Функції для завантаження мережевих об'єктів Функції для аналізу URL Можливості urllib2 XML-RPC сервер	Лекція	[1], [2], [10],	2	кінець поточного тижня
11	Лаб.9. ООП: створення класа та об'єктів-екземплярів класу	Лабораторна робота		2	кінець поточного тижня
12	Робота з базою даних. Основні поняття реляційної СУБД Що таке DB-API 2 Опис DB API 2.0 Інтерфейс	Лекція	[1], [2], [4], [8],	2	кінець поточного тижня

	модуля Об'єкт-з'єднання Об'єкт-курсор Робота з базою даних з Python-додатки Знайомство з СУБД Створення бази даних Наповнення бази даних Вибірки з бази даних Інші СУБД і Python				
12	Лаб.10. Наслідування атрибутів класу в ООП.	Лабораторна робота		2	кінець поточного тижня
13	Багатопотокові обчислення. Про потоках управління Приклад багатопотокової програми Функції модуля threading Клас Thread Таймер Замки Семафори Події Умови Черга Модуль thread Візуалізація роботи потоків	Лекція	[1], [2], [4], [8],	2	кінець поточного тижня
13	Лаб.11. ООП: розробка сховища даних.	Лабораторна робота		2	кінець поточного тижня
14	Створення додатків з GUI. Огляд графічних бібліотек Про графічному інтерфейсі Основи Tk Класи віджетів Події Створення і конфігурація віджета Віджет тексту фіксованої Менеджери розташування Зображення в Tkinter Графічне додаток на Tkinter	Лекція	[1], [2], [4], [8],	2	кінець поточного тижня
14	Лаб.12. Розробка додатків з графічним інтерфейсом на основі бібліотеки tkinter	Лабораторна робота		2	кінець поточного тижня
15	Інтеграція Python з іншими мовами програмування С API Написання модуля розширення Приклад вбудовування інтерпретатора в програму на С Використання SWIG Простий приклад використання SWIG Інтеграція Python і інших систем програмування	Лекція	[1], [2], [4], [10]	2	кінець поточного тижня
15	Модуль 3	Модуль		2	кінець поточного тижня
16	Пристрій інтерпретатора мови Python Лексичний аналіз Синтаксичний аналіз Отримання байт-коду Вивчення байт-коду Налаштування Профайлер Оптимізація Дослідження об'єкта	Лекція	[4], [7],	2	кінець поточного тижня
16	Підсумкове заняття	Лабораторна робота		2	кінець поточного тижня