

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**  
**Львівський національний університет імені Івана Франка**  
**Механіко-математичний факультет**  
**Кафедра математичного моделювання соціально-економічних процесів**

**Затверджено**  
на засіданні кафедри математичного  
моделювання соціально-економічних  
процесів Львівського національного  
університету імені Івана Франка  
(протокол № 1 від 29 серпня 2022 року)

Завідувач кафедри



проф. Сеньо П.С.

Силабус з навчальної дисципліни

**“Інформатика і програмування (Python)”**

що викладається в межах ОПП *“Комп’ютерна алгебра, криптологія і теорія ігор”*, *“Комп’ютерний аналіз математичних моделей”*, *“Математика. Математична економіка та економетрика”*,

першого (бакалаврського) рівня вищої освіти для здобувачів

із спеціальності 111 – Математика

**Львів 2022**

<b>Назва дисципліни</b>	<b>Інформатика і програмування (Python)</b>
<b>Адреса викладання дисципліни</b>	Львівський національний університет імені Івана Франка, вул. Університетська 1, м. Львів, Україна, 79000
<b>Факультет та кафедра, за якою закріплена дисципліна</b>	Факультет прикладної математики та інформатики, кафедра математичного моделювання соціально-економічних процесів
<b>Галузь знань, шифр та назва спеціальності</b>	Галузь знань: 11 Математика і статистика Спеціальність: 111 Математика
<b>Викладачі дисципліни</b>	<b>Ярошко Світлана Михайлівна</b> , к. ф.-м. н., доцент каф. математичного моделювання соціально-економічних процесів (ММСЕП); <b>Добуляк Олександра Петрівна</b> , к.ф.-м.н., доцент каф. ММСЕП; <b>Костенко Світлана Борисівна</b> , к.ф.-м.н., доцент каф. ММСЕП; <b>Мокрицький Тарас Володимирович</b> , асист. каф. ММСЕП
<b>Контактна інформація викладачів</b>	<a href="mailto:svitlana.yaroshko@lnu.edu.ua">svitlana.yaroshko@lnu.edu.ua</a> <a href="mailto:lesia.dobuliak@lnu.edu.ua">lesia.dobuliak@lnu.edu.ua</a> <a href="mailto:svitlana.kostenko@lnu.edu.ua">svitlana.kostenko@lnu.edu.ua</a> <a href="mailto:taras.mokrytskyi@lnu.edu.ua">taras.mokrytskyi@lnu.edu.ua</a>
<b>Консультації з питань навчання по дисципліні відбуваються</b>	Один раз на тиждень згідно з оприлюдненим розкладом консультацій викладача. Можливі онлайн-консультації в середовищі Microsoft Teams. Для погодження часу онлайн-консультацій треба писати на електронну пошту викладача.
<b>Сторінка курсу</b>	<a href="https://ami.lnu.edu.ua/employee/yaroshko-svitlana">https://ami.lnu.edu.ua/employee/yaroshko-svitlana</a>
<b>Інформація про дисципліну</b>	Курс «Інформатика і програмування (Python)» є нормативною навчальною дисципліною зі спеціальності 111 – Математика для освітніх програм “Комп’ютерна алгебра, криптологія і теорія ігор”, “Комп’ютерний аналіз математичних моделей”, “Математика. Математична економіка та економетрика”, яку викладають у першому та другому семестрах в обсязі 6 кредитів (за Європейською кредитно-трансферною системою ECTS).
<b>Коротка анотація дисципліни</b>	Курс фокусується на особливостях моделі даних мови Python та на бібліотечних засобах їхнього аналізу та візуалізації.
<b>Мета та цілі дисципліни</b>	<b>Мета:</b> формування компетенцій, необхідних для ефективного використання бібліотек мови Python з метою вирішення прикладних та наукових задач. <b>Цілі:</b> ознайомлення з технологіями опрацювання та візуалізації даних засобами бібліотек мови Python, оволодіння навиками побудови та програмування алгоритмів розв’язання прикладних задач у імперативному, процедурному та об’єктно-орієнтованому стилях.
<b>Література для вивчення дисципліни</b>	<i>Основна:</i> <a href="#">Маттес Е. Пришвидшений курс Python.</a> – Львів : ВСЛ, 2021. Селіверстов Р., Мельничин А. Основи програмування мовою Python: навч. посібник. – Львів : ЛНУ імені Івана Франка, 2020. The Python Tutorial. – <a href="https://docs.python.org/3/tutorial/index.html">https://docs.python.org/3/tutorial/index.html</a> Lutz M. Learning Python, 5th Edition. – O’Reilly Media, 2013. Lambert K. A. Fundamentals of Python: First Programs, 2nd Edition. – Cengage, 2019. NumPy. – <a href="http://numpy.org">http://numpy.org</a> .

	<p>Pandas. – <a href="http://pandas.pydata.org">http://pandas.pydata.org</a> .  Matplotlib. – <a href="http://matplotlib.org">http://matplotlib.org</a> .  <i>Додаткова:</i>  Wentworth P., Elkner J., Downey A., Meyers C. How to Think Like a Computer Scientist: Learning with Python 3. – Green Tea Press, 2018.  Python Tricks: The Book. — Dan Bader, 2017.  Sweigart A. Automate the Boring Stuff with Python: Practical Programming for Total Beginners. – No Starch Press, 2014.  McKinney W. Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython, 2nd Edition. – O’Reilly Media, 2018.</p>
<b>Обсяг курсу</b>	<p><b>Всього:</b> 180 годин, з них 48 годин лекцій, 48 годин лабораторних занять, 84 години самостійної роботи. 6 кредитів ЄКТС.  <b>1-ий семестр:</b> 75 годин, з них 16 годин лекцій, 16 годин лабораторних занять, 43 години самостійної роботи. 2,5 кредитів ЄКТС.  <b>2-ий семестр:</b> 105 годин, з них 32 години лекцій, 32 години лабораторних занять, 41 година самостійної роботи. 3,5 кредитів ЄКТС.</p>
<b>Очікувані результати навчання</b>	<p>Після завершення цього курсу студент буде <i>знати:</i></p> <ul style="list-style-type: none"> <li>• синтаксис, основні конструкції та типи даних Python;</li> <li>• принципи імперативного, процедурного, функціонального та об’єктно-орієнтованого програмування мовою Python;</li> <li>• класичні та спеціалізовані алгоритми обробки та візуалізації даних, які реалізовані у популярних бібліотеках Python.</li> </ul> <p><i>вміти:</i></p> <ul style="list-style-type: none"> <li>• реалізовувати класичні алгоритми мовою Python;</li> <li>• використовувати переваги об’єктно-орієнтованого програмування під час написання програм;</li> <li>• розробляти графічні інтерфейси;</li> <li>• представляти, структурувати, обробляти та візуалізувати дані у комп’ютерних системах;</li> <li>• застосовувати одержані знання для розв’язання конкретних професійних задач.</li> </ul>
<b>Компетентності</b>	<p><b>Загальні (ЗК) і спеціальні компетентності (СК):</b>  ЗК-6 Навички використання інформаційних і комунікаційних технологій;  ЗК-7 Здатність учитися і оволодівати сучасними знаннями;  СК-7 Здатність застосовувати чисельні методи для дослідження математичних моделей;  СК-9 Здатність застосовувати спеціалізовані мови програмування та пакети прикладних програм;  СК-10 Здатність використовувати обчислювальні інструменти для чисельних і символічних розрахунків.</p> <p><b>Програмні результати навчання:</b>  РН-5 Мати навички використання спеціалізованих програмних засобів комп’ютерної та прикладної математики і використовувати інтернет-ресурси;</p>

	<p>PH-20 Розв'язувати основні математичні задачі аналізу даних; застосовувати базові загальні математичні моделі для специфічних ситуацій, мати навички управління інформацією, і застосування комп'ютерних засобів статистичного аналізу даних;</p> <p>PH-21 Розв'язувати типові задачі математичного аналізу, алгебри, диференціальних та інтегральних рівнянь, оптимізації за допомогою чисельних методів.</p>
<b>Ключові слова</b>	алгоритм, скрипт, тип даних, інструкція, функція, об'єкт, клас, поліморфізм, наслідування, інкапсуляція
<b>Формат курсу</b>	Проведення лекцій, лабораторних робіт та консультацій (очно і дистанційно на платформі Microsoft Teams ), самостійна робота студента
<b>Теми</b>	Див. схему курсу
<b>Підсумковий контроль, форма</b>	Диференційований залік у кінці першого семестру, екзамен у кінці другого семестру
<b>Пререквізити</b>	Для вивчення курсу студенти потребують базових знань з шкільного курсів математики та інформатики. Одночасно з вивченням курсу студенти проходять у другому семестрі навчальну обчислювальну практику, впродовж якої виконують завдання з програмування, поглиблюють набуті в курсі знання та удосконалюють навички.
<b>Навчальні методи та техніки, які будуть використовуватися під час викладання курсу</b>	Лекції з презентаціями; лабораторні заняття у вигляді виконання практичних завдань; самостійне опрацювання навчальних матеріалів, розміщених у хмарних сховищах (Moodle, Microsoft Teams); обговорення тем та консультації в середовищах Microsoft Teams, Skype тощо.
<b>Необхідне обладнання</b>	Для проведення лекцій: комп'ютер, проектор, доступ до мережі Інтернет. Для проведення лабораторних занять та виконання завдань: комп'ютер, ОС Windows/Linux, середовище програмування.
<b>Критерії оцінювання (окремо для кожного виду навчальної діяльності)</b>	Оцінювання проводиться за 100-бальною шкалою. Залік у кінці першого семестру є підсумком виконання індивідуальних та лабораторних завдань протягом семестру. У другому семестрі 50 балів нараховують за виконання лабораторних робіт, ще 50 балів – за виконання екзаменаційного завдання. Упродовж семестру студент виконує не менше 8 лабораторних робіт, кожна з яких оцінюють від 2 до 10 балів залежно від складності. Оцінка за екзаменаційне завдання складається з двох частин: 20 балів за засвоєння теоретичного матеріалу (тест) та 30 балів за написання комп'ютерних програм. Студенти зобов'язані дотримуватися усіх строків визначених для виконання усіх видів робіт, передбачених курсом. Звіти про виконання лабораторних робіт завантажують у відповідне хмарне сховище. Обов'язковим є захист лабораторних робіт. Виявлення ознак академічної недобросовісності в роботі студента (списування, здавання чужих звітів, несамостійність виконання завдань) є підставою для її незарахування викладачем.
<b>Питання до екзамену</b>	Тематика теоретичної (тестової) частини: 1. Базові типи об'єктів Python.

2. Особливості та наслідки динамічної типізації.
  3. Операції над числовими типами.
  4. Операції над рядками.
  5. Упорядковані колекції об'єктів.
  6. Невпорядковані колекції об'єктів.
  7. Умовні інструкції та цикли.
  8. Основи процедурного програмування.
  9. Області видимості.
  10. Спеціальні режими співставлення аргументів.
  11. Анонімні функції.
  12. Оперування функціями як об'єктами.
  13. Засоби функціонального програмування в Python.
  14. Модулі та пакети.
  15. Класи і ООП.
  16. Перевантаження операторів.
  17. Наслідування.
  18. Обробка винятків.
  19. Декоратори.
  20. Засоби візуалізації даних matplotlib
  21. Багатомірні масиви NumPy
  22. Засоби аналізу даних Pandas
- Для успішного складання практичної частини (програмна реалізація) потрібно вміти:
1. Використовувати інструкції галуження та циклів, зокрема, з інструкціями break, continue і блоком else в циклах.
  2. Використовувати стандартні контейнери: рядки, кортежі, списки, словники, множини.
  3. Оголошувати та використовувати функції з позиційними та іменованими аргументами, параметрами зі значеннями за замовчуванням, у тому числі і зі змінною кількістю аргументів.
  4. Оперувати функціями як об'єктами: оголошувати функції всередині інших функцій, передавати функції аргументами інших функцій, додавати атрибути функціям тощо.
  5. Використовувати лямбда-функції.
  6. Оголошувати класи, визначаючи конструктори, методи, статичні атрибути, перевантажуючи оператори.
  7. Наслідувати класи, перевизначати методи.
  8. Оголошувати, запускати та опрацьовувати винятки, розпізнаючи тип винятку, отримавши доступ до його даних та гарантуючи виконання певних дій незалежно від виникнення винятків.
  9. Взаємодіяти з файловою системою комп'ютера: знаходити, створювати, переіменувати, вилучати файли та каталоги.
  10. Завантажувати дані з файлів (текстових і двійкових), серіалізувати дані.
  11. Будувати основні типи графіків засобами бібліотеки matplotlib, налаштовувати їх параметри.
  12. Використовувати засоби бібліотеки numpy для опрацювання числових масивів.
  13. Використовувати графічні елементи керування (widgets) для створення інтерактивного інтерфейсу користувача.

	14. Використовувати засоби бібліотеки pandas для опрацювання таблиць гетерогенних даних (завантаження таблиць, об'єднання таблиць, заповнення пропусків, вибірка, фільтрування, використання методів агрегації, застосування функцій користувача).
<b>Опитування</b>	Анкету-оцінку з метою оцінювання якості курсу буде надано по завершенню курсу.

<b>Схема курсу</b>				
Тиждень	Тема, план, короткі тези	Форма діяльності	Тривалість, год	Термін виконання
1-4	<i>Модель даних, базові типи, оператори та інструкції Python: Режими виконання коду. Основні прийоми роботи в Jupyter Notebook. Коментування коду, довідка Python. Особливості числових типів. Введення-виведення даних. Стандартна бібліотека Python. Імпорт.</i>	<i>лекція</i>	<i>4</i>	
	<i>Програмна реалізація лінійних алгоритмів</i>	<i>лабораторне заняття</i>	<i>4</i>	<i>наступне лаб. заняття</i>
5-8	<i>Розгалуження та цикли. Розгалуження в алгоритмах. Логічні вирази. Умовні конструкції if. Функція range(). Цикли for і while. Операції break і continue. Блок else.</i>	<i>лекція</i>	<i>4</i>	
	<i>Умовні конструкції та цикли</i>	<i>лабораторне заняття</i>	<i>4</i>	<i>наступне лаб. заняття</i>
9-12	<i>Рядки та файли: Літерали рядків. Символи. Керівні символи. Індеси та зрізи. Конкатенація та повторення рядків. Форматування рядків. Функції для роботи з рядками та методи рядків. Перевірка на входження та посимвольний обхід рядка. Режими відкриття файлів. Операції з файлами. Інструкція with/as. Шляхи.</i>	<i>лекція</i>	<i>4</i>	
	<i>Програмування з використанням рядків і файлів</i>	<i>лабораторне заняття</i>	<i>4</i>	<i>наступне лаб. заняття</i>
13-16	<i>Колекції: Списки. Кортежі. Словники. Множини. Доступ до елементів. Перевірка на</i>	<i>лекція</i>	<i>4</i>	

	<i>входження та поелементний обхід. Основні функції та методи. Генерування колекцій.</i>			
	<i>Програмування з використанням колекцій</i>	<i>лабораторне заняття</i>	4	<i>наступне лаб. заняття</i>
17-20	<i>Функції: Декомпозиція. Функції та аргументи. Створення та виклик функції. Інструкція return. "Мертвий" код. Області видимості. Локальні та глобальні змінні. Вкладення функцій. Лямбда-функції. Позиційні та іменовані аргументи. Значення за замовчуванням. Передавання довільної кількості аргументів. Розпаковування аргументів. Спеціальні режими зіставлення аргументів. Рекурсивні функції. Поняття про обробку даних засобами питру та візуалізація за допомогою matplotlib</i>	<i>лекція</i>	8	
	<i>Процедурне програмування. Декомпозиція</i>	<i>лабораторне заняття</i>	8	<i>наступне лаб. заняття</i>
21-24	<i>Класи та об'єкти (екземпляри). Атрибути та методи, особливості інкапсуляції. Створення екземплярів класу. Методи __init__() та __str__(). Особливості керування доступом до атрибутів; property. Змінні та методи класу. Статичні методи. Перевантаження операторів. Наслідування класів, особливості множинного наслідування. Перевизначення методів. Виклик методів базового класу.</i>	<i>лекція</i>	8	
	<i>Об'єктно-орієнтоване програмування</i>	<i>лабораторне заняття</i>	8	<i>наступне лаб. заняття</i>
25, 26	<i>Прийоми ефективного використання вбудованих колекцій; ітератори, генератори. Інструкція assert. Модульне тестування: unittest.</i>	<i>лекція</i>	4	
	<i>Тестування програм</i>	<i>лабораторне заняття</i>	4	<i>наступне лаб. заняття</i>
27, 28	<i>Основи програмування графічних інтерфейсів. Графічний інтерфейс</i>	<i>лекція</i>	4	

	<i>користувача. Модуль tkinter. Базове вікно. Елементи управління (мітки, кнопки, текстові поля і області, прапорці опцій, перемикачі). Обробка подій.</i>			
	<i>Програмування графічних інтерфейсів</i>	<i>лабораторне заняття</i>	<i>4</i>	<i>наступне лаб. заняття</i>
<i>29</i>	<i>Візуалізація даних засобами matplotlib.</i>	<i>лекція</i>	<i>2</i>	
	<i>Візуалізація даних з matplotlib.</i>	<i>лабораторне заняття</i>	<i>2</i>	<i>наступне лаб. заняття</i>
<i>30, 31</i>	<i>Матрично-векторні обчислення засобами NumPy.</i>	<i>лекція</i>	<i>4</i>	
	<i>Операції над матрицями в NumPy.</i>	<i>лабораторне заняття</i>	<i>2</i>	<i>наступне лаб. заняття</i>
<i>32</i>	<i>Аналіз даних засобами Pandas.</i>	<i>лекція</i>	<i>2</i>	
	<i>Статистичний аналіз даних</i>	<i>лабораторне заняття</i>	<i>2</i>	<i>наступне лаб. заняття</i>
	<i>Підсумкове заняття</i>	<i>лабораторне заняття</i>	<i>2</i>	