

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 РКС
факультету електроніки та комп'ютерних
технологій

Львівського національного університету
імені Івана Франка

(протокол № ____ від 31 серпня 2020 р.)

Завідувач кафедри _____

Силабус з навчальної дисципліни
«Мова програмування Python»,
що викладається в межах ОПП «Інформаційні системи та
технології» першого (бакалаврського) рівня вищої освіти для
здобувачів з спеціальності
126 «Інформаційні системи та технології»

Назва дисципліни	Мова програмування Python
Адреса викладання дисципліни	м. Львів, вул. Тарнавського, 107
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	Свелеба Сергій Андрійович, докт. фіз.-мат. наук, ст.наук.співр, професор
Контактна інформація викладачів	serhiy.sveleba@lnu.edu.ua, Свелеба Сергій Андрійович – Факультет електроніки та комп'ютерних технологій (lnu.edu.ua)
Консультації з питань навчання по дисципліні відбуваються	Консультації в день проведення лекційних занять (за попередньою домовленістю). Також можливі он-лайн консультації через MS Teams. Для погодження часу он-лайн консультацій слід писати на електронну пошту викладача.
Сторінка дисципліни	http://194.44.208.156/moodle/course/view.php?id=51
Інформація про дисципліну	Дисципліна «Мова програмування Python» є вибірковою дисципліною з спеціальності 126 Інформаційні системи та технології для освітньої програми «Інформаційні системи та технології», яка викладається в 6 семестрі в обсязі 5,5 кредитів (за Європейською Кредитно-Трансферною Системою ECTS).
Коротка анотація дисципліни	Розглядаються базові поняття структурного програмування на мові Python 3.X. <i>Метою навчальної дисципліни</i> є формування у студентів базових понять структурного програмування на Python 3.X, вміння застосовувати їх на практиці.
Мета та цілі дисципліни	Метою вивчення вибіркової дисципліни «Мова програмування Python» 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.
Література для вивчення дисципліни	Основна література: 1. Абрамов С. А. Начала информатики. М.: «Наука», 1990. – 256 с. 2. Абрамов С. А. и др. Задачи по программированию. – М.: «Наука», 1988. – 224 с. 3. Аленский Н. А. и др. Задачи и методические рекомендации по программированию. – Мн.: БГУ, 1990. – 67 с. 4. Вирт Н. Алгоритмы и структуры данных. – М., 1989, – 360 с. 5. Златопольский Д. М. Сборник задач по программированию. – 3-е изд., перераб. и доп. – СПб.: БХВ–Петербург, 2011. – 304 с.: ил. – (ИиИКТ) 6. Иванова Г.С. Основы программирования: Учебник для вузов. – 4-е изд., стер. – М.: Изд-во МГТУ им. Н.Э. Баумана, 2007.-416 с.: ил. (Сер. «Информатика в техническом университете».) 7. Лутц М. Изучаем Python. 4-е издание. – Пер. с англ. – СПб.: СимволПлюс, 2011. – 1280 с. 8. Лутц М. Программирование на Python, том 1, 4-е издание. – Пер. с англ. – СПб.: Символ-Плюс, 2011. – 992 с. , ил. [гл. 7–10] 9. Руководство пользователя OpenOffice.org 2. – Издательство: БХВ–Петербург, 2007. – 320 с. 10. Сборник лабораторных работ с примерами решения задач по

	<p>алгоритмизации и программированию на языке Си: Учебно-методическое пособие для студентов высших технических учебных заведений / Кравчук А.И., Кравчук А.С. – Мн.: Технопринт, 2002, 111 с.</p> <p>11. Шапошникова С. Основы программирования на Python. – Лаборатория юного линуксоида, 2011г. – 44 с.</p> <p>12. Седер Наоми C28 Python. Экспресс-курс. 3-е изд. — СПб.: Питер, 2019. — 480 с.:</p> <p>13. Федоров, Д. Ю. Программирование на языке высокого уровня Python : учебное пособие для прикладного бакалавриата / Д. Ю. Федоров. — М. : Издательство Юрайт, 2017. — 126 с.</p> <p>14. Хахаев И. А. Практикум по алгоритмизации и программированию на Python: / И.А.Хахаев — М. : Альт Линукс, 2010. — 126с.</p> <p>15. Прохоренок Н. А. Python. Самое необходимое. — СПб.: БХВ-Петербург, 2011. — 416 с.:</p> <p>16. Мэтиз Эрик M54 Изучаем Python. Программирование игр, визуализация данных, веб-приложения. — СПб.: Питер, 2017. — 496 с.:</p>
Обсяг курсу	165 години аудиторних занять. З них 32 години лекцій, 32 години лабораторних робіт та 101 година самостійної роботи
Очікувані результати навчання	<p>Після завершення цього курсу студент буде:</p> <ul style="list-style-type: none"> Знати 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 (інструкції <i>if</i>, <i>while</i>, <i>for</i>); принципи побудови функцій; модульний принцип організації програми;основи побудови сховищ даних Вміти використовувати базові поняття структурного програмування для розв'язання різних задач; створювати і обробляти структури, масиви структур, найпростішу обробку файлів, створювати і використовувати функції.
Ключові слова	Цикли, масиви, вбудовані функції, функції користувача, класи, атрибути класу, об'єкти класу
Формат курсу	Очний
	Проведення лекцій, лабораторних робіт та консультації для кращого розуміння тем
Теми	Див. СХЕМА КУРСУ
Підсумковий контроль, форма	Залік в кінці семестру
Пререквізити	Для вивчення курсу студенти потребують базових знань з дисциплін «Вища математика», «Дискретна математика», «Алгоритми та структури даних», «Математичний аналіз», «Диференціальні рівняння» та «Обчислювальна техніка і програмування» та «Обчислювальна техніка і програмування»
Навчальні методи та техніки, які будуть використовуватися під час викладання курсу	Презентація, лекції, лабораторні роботи, обговорення, дискусія.
Необхідне обладнання	Мультимедіа, платформа Moodle, комп'ютерне програмне забезпечення
Критерії оцінювання	Оцінювання проводиться упродовж семестру за 100-бальною шкалою.

<p>(окремо для кожного виду навчальної діяльності)</p>	<p>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. • контрольні заміри (3 модулі): 56% семестрової оцінки; максимальна кількість балів 56. <p>Загалом упродовж семестру 100 балів.</p> <hr/> <p>Контрольні заміри проводяться у формі тестових завдань. Академічна доброчесність: Очікується, що лабораторні та контрольні роботи студентів будуть їх оригінальними дослідженнями чи міркуваннями. Відсутність посилань на використані джерела, фабрикування джерел, списування, втручання в роботу інших студентів становлять, але не обмежують, приклади можливої академічної недоброчесності. Виявлення ознак академічної недоброчесності в роботі студента 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 і лабораторні заняття курсу. Студенти мають інформувати викладача про неможливість відвідати заняття. Студенти зобов'язані дотримуватися усіх термінів визначених для виконання усіх 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 на поточному тестуванні, самостійній роботі та бали підсумкового тестування. При цьому обов'язково враховуються присутність на заняттях та активність студента під час лабораторного 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.</p> <p>Жодні форми порушення академічної доброчесності не толеруються.</p>
<p>Питання до контрольних робіт</p>	<p>Перелік питань та завдань для проведення підсумкової оцінки знань певних тем до контрольних робіт розміщені на веб-сторінці.</p>
<p>Опитування</p>	<p>Анкету-оцінку з метою оцінювання якості курсу буде надано по завершенню курсу.</p>

СХЕМА КУРСУ

Тиж.	Тема, план, короткі тези	Форма діяльності (заняття)	Література. Ресурси в Інтернеті	Завдання (лабораторна робота), год	Термін виконання
1	<p>Алгоритми послідовної (лінійної) структури. Числа та операції над ними. Введення даних з клавіатури</p> <p>Алгоритми послідовної (лінійної) структури, з процедурами запуску програм, які реалізують ці алгоритми на мові Python; запуск програми користувачем; числа та операції над ними;</p>	Лекція	1, 2, 5	Розробка блок-схеми алгоритму розв'язання задачі. Обчислити вирази.	3 тиж. семестру

	введення даних з клавіатури.				
2, 3	Алгоритми розгалуженої структури (інструкція if) Алгоритми розгалуженої структури та їх реалізація. ;динамічна типізація; умовна інструкція if; реалізація обробки помилок (виключень у вигляді рядків): за допомогою інструкції try; перевірка змісту рядку введення за допомогою методу isdigit.	Лекція	1, 2, 5, 7, 8	а)Алгоритми розгалуженої структури (інструкція if). б)Алгоритми циклічної структури (інструкція while). в) Алгоритми циклічної структури (інструкція for).	4 тиж. семестру
4	Алгоритми циклічної структури (інструкція while) Алгоритми циклічної структури, їх реалізацією на мові Python.; оператор циклу while; тип даних «рядки».	Лекція	1, 2, 5, 8	а) Функції користувача. Наближене обчислення функцій. б) Рекурсивні функції. Документування коду.	5 тиж. семестру
5	Алгоритми циклічної структури (інструкція for) Алгоритми циклічної структури на основі оператору циклу for, модуль itertools (функція itertools.count), функція range, ітераційні процеси.	Лекція	2, 5, 7,9, 10	Рядки символів. Завдання 1 - обробка рядків. Завдання 2 – обробка речення.	7 тиж. семестру
6,7	Списки: одновимірні та двовимірні масиви. Генерація випадкових чисел Одновимірні та двовимірні масиви, структура організації масивів та способи доступу до їх елементів, списки, масиви даних, процедура генерації випадкових чисел (модуль random)	Лекція	2, 5, 7, 11	Списки: одновимірні та двовимірні масиви. Генерація випадкових чисел. Завдання 1. Одновимірні масиви (вектори). Завдання 2. Двовимірні масиви (матриці).	8 тиж. семестру
8	Методи. Сортування. Присвоєння Методи додавання елемента « ?» у список; додавання списку у список. додавання списку елементів у вказану позицію; визначення зсуву елемента за заданим значенням; підрахунок кількості елементів; сортування; зміна порядку слідування елементів; зменшення списку; видалення елемента; видалення групи елементів; присвоєння за індексом; присвоєння зрізу значень; генератори списків.	Лекція	1, 2, 5	Рядки символів. Множини.	9 тиж. семестру
9	Рядки символів. Множини (Множина.; рядки символів; обробка рядків; використання множин; пошук перетину двох послідовностей у вигляді рядків; використання множини для визначення перетину послідовностей у вигляді рядків; створення множин на основі генераторів множин; використання конструктору frozenset; операції над множинами, записані у вигляді	Лекція	1, 5, 8, 10	Словник. Кортеж	10 тиж. семестру

	бінарних операцій.				
10, 11	Функції користувача. Наближене обчислення функцій Основні складові функцій, оголошення та опис функцій; оператор форматування %; інструкції, які використовують при створенні функцій.	Лекція	2, 3, 5, 11, 12, 13	Файли даних. Модульний принцип організації програми. Завдання 1: бінарні файли. Завдання 2: текстові файли.	12 тиж. семестру
12	Рекурсивні функції. Документування коду Організація рекурсивних функцій користувача, методики визначення та практичного застосування рекурсивних функцій у програмах, ефективний засіб реалізації циклічних алгоритмів, документування коду, вбудовані рядки документування, Lambda-вирази.	Лекція	2, 4, 5	Файли даних. Модульний принцип організації програми. Завдання 1: бінарні файли. Завдання 2: текстові файли.	13 тиж. семестру
13	Словник. Кортеж Словники, кортежі мови Python, створення словників, операції зі словниками, використання collections.Counter, використання lambda-виразів, літерали словників та операції, методи словників, літерали кортежів та операції.	Лекція	1, 2, 4	Класи та об'єкти – екземплярів класу.	14 тиж. семестру
14	Файли даних. Модульний принцип організації програми Основні положення роботи з файлами на мові Python. Бінарні та текстові файли, модульний принцип організації програми. Файл. Модуль. Відкриття файлу для зчитування. Використання функції os.path.join для побудови шляху до файлу. Запис даних у текстовий файл. Використання оператора with для закриття файлу. Відкриття текстового файлу для зчитування з вказівкою кодування __file__. Відкриття файлу для зчитування та запису. Відкриття файлу для до запису. Перезапис файлу. Використання файлового об'єкта io.StringIO. Використання бінарного файлу. Використання json. Серіалізація – десеріалізація. Серіалізація за допомогою pickle. Декоратор.	Лекція	2, 5, 6, 9, 14, 15	Побудова Фреймової моделі. Наслідування атрибутів класу в ООП.	15 тиж. семестру
15, 16	Побудова Фреймової моделі. Наслідування атрибутів класу в ООП. Успадкування. Інкапсуляція. Поліморфізм. ООП: створення класу та об'єктів-екземплярів класу. Наслідування атрибутів класу в ООП. ООП: розробка сховища даних	Лекція	2, 3, 4, 5, 12, 13, 14,16	ООП: розробка сховища даних	16 тиж. семестру

