

ПОГЛИБЛЕНИЙ КУРС ПРОГРАМУВАННЯ PYTHON

СИЛАБУС

Шифр і назва спеціальності	121 Інженерія програмного забезпечення, 122 Комп'ютерні науки, 126 Інформаційні системи та технології	Інститут / факультет	Факультет комп'ютерних наук і програмної інженерії
Назва програми	«Інженерія програмного забезпечення» (Innovation Campus)	Кафедра	Кафедра програмної інженерії та інформаційних технологій управління
Тип програми	Освітньо-професійна	Мова навчання	Українська, Англійська

Викладач

ПІБ, електронна пошта

Козуля Марія Михайлівна, mariia.kozulia@khpri.edu.ua



Кандидат технічних наук (21.06.01 - екологічна безпека), доцент кафедри програмної інженерії та інформаційних технологій управління. Досвід роботи – з 2016 року. Автор (співавтор) понад 75 наукових та навчально-методичних публікацій (h-index= 6, i10-index= 2 in Google Scholar - <https://scholar.google.ru/citations?user=tRyBDzQAAAAJ&hl=ru>; ORCID <https://orcid.org/0000-0002-4090-8481>). Основні курси: «Основи програмування Python» (лекції та лабораторні заняття), «Поглиблений курс програмування Python» (лекції та лабораторні заняття), «Теорія прийняття рішень» (лекції та лабораторні заняття), «Моделі та методи теорії прийняття рішень» (лекції та лабораторні заняття), «Грін комп'ютинг» (лекції та лабораторні заняття).

Загальна інформація про курс

Анотація	Курс «Поглиблений курс програмування Python» є навчальною дисципліною з циклу вибіркової підготовки за спеціальністю 121 «Інженерія програмного забезпечення», 122 Комп'ютерні науки, 126 Інформаційні системи та технології. Вона викладається у третьому семестрі в обсязі 180 годин (6 кредити ECTS), зокрема: лекції – 32 годин, лабораторні заняття – 32 години, самостійна робота – 116 години. Індивідуальних завдань не передбачено. Вивчення дисципліни завершується заліком.
Цілі курсу	Формування у студентів теоретичних та практичних знань з оволодіння практичними навичками: переформатування, очистки та обробки даних на Python; практичного використання бібліотек Matplotlib, Pandas, NumPy, IPython для аналізу даних; роботи з базами даних на мові Python; створення віконних додатків за допомогою бібліотек Tkinter та PyQt5.
Формат	Лекції, лабораторні заняття. Поточний контроль – лабораторні роботи, проміжний модульний контроль. Підсумковий контроль – залік.
Семестр	4

Обсяг (кредити) / Тип курсу (обов'язковий / вибіркового)	б/ Вибірковий	Лекції (години)	32	Лабораторні заняття (години)	32	Самостійна робота (години)	116
Програмні 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-K15. Здатність розробляти архітектури, модулі та компоненти програмних систем.

121-K16. Здатність формулювати та забезпечувати вимоги щодо якості програмного забезпечення у відповідності з вимогами замовника, технічним завданням та стандартами.

121-K19. Володіння знаннями про інформаційні моделі даних, здатність створювати програмне забезпечення для зберігання, видобування та опрацювання даних.

121-K26 Здатність до алгоритмічного та логічного мислення

122-3K1. Здатність до абстрактного мислення, аналізу та синтезу.

122-3K2. Здатність застосовувати знання у практичних ситуаціях.

122-3K6. Здатність вчитися і оволодівати сучасними знаннями.

122-3K7. Здатність до пошуку, оброблення та аналізу інформації з різних джерел.

122-3K12. Здатність оцінювати та забезпечувати якість виконуваних робіт.

122-СК3. Здатність до логічного мислення, побудови логічних висновків, використання формальних мов і моделей алгоритмічних обчислень, проектування, розроблення й аналізу алгоритмів, оцінювання їх ефективності та складності, розв'язності та нерозв'язності алгоритмічних проблем для адекватного моделювання предметних областей і створення програмних та інформаційних систем.

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.

126-K31. Здатність до абстрактного мислення, аналізу та синтезу.

126-K32. Здатність застосовувати знання у практичних ситуаціях.

126-K3 5. Здатність вчитися і оволодівати сучасними знаннями.

126-K3 6. Здатність до пошуку, оброблення та узагальнення інформації з різних джерел.

126-K3 8. Здатність оцінювати та забезпечувати якість виконуваних робіт.

126-КС 4. Здатність проектувати, розробляти та використовувати засоби реалізації інформаційних систем, технологій та інфокомунікацій (методичні, інформаційні, алгоритмічні, технічні, програмні та інші).

126-КС 6. Здатність використовувати сучасні інформаційні системи та технології (виробничі, підтримки прийняття рішень, інтелектуального аналізу даних та інші), методики й техніки кібербезпеки під час виконання функціональних завдань та обов'язків.

126-КС 8. Здатність управляти якістю продуктів і сервісів інформаційних систем та технологій протягом їх життєвого циклу.

Результати навчання	Методи викладання та навчання	Форми оцінювання (поточне оцінювання CAS, підсумкове оцінювання FAS)
<p>121-ПР01. Аналізувати, цілеспрямовано шукати і вибрати необхідні для вирішення професійних завдань інформаційно-довідникові ресурси і знання з урахуванням сучасних досягнень науки і техніки.</p> <p>121-ПР07. Знати і застосовувати на практиці фундаментальні концепції, парадигми і основні принципи функціонування мовних, інструментальних і обчислювальних засобів інженерії програмного забезпечення.</p> <p>121-ПР11. Вибирати вихідні дані для проектування, керуючись формальними методами опису вимог та</p>	<p>У процесі викладання передбачено застосування таких начальних технологій, як: лекції, лабораторні роботи, презентації, що розвивають комунікативні та лідерські навички, самостійна робота з літературними джерелами, змішані форми навчання з використанням дистанційних платформ</p>	<p>Поточне оцінювання CAS: Оцінювання роботи студентів у процесі лабораторних занять Проміжний модульний контроль</p> <p>Підсумкове оцінювання FAS: Залік</p>

модельовання.

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-ПР17. Вміти застосовувати методи компонентної 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-ПР9. Розробляти програмні моделі предметних середовищ, вибирати парадигму програмування з позицій зручності та якості застосування для реалізації методів та алгоритмів розв'язання задач в галузі комп'ютерних наук.

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

122-ПР13. Володіти мовами системного програмування та методами розробки програм, що взаємодіють з компонентами комп'ютерних систем, знати мережні технології, архітектури комп'ютерних мереж, мати практичні навички технології адміністрування комп'ютерних мереж та їх програмного забезпечення.

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.

126-ПР 3. Використовувати базові знання інформатики й сучасних інформаційних систем та технологій, навички програмування, технології безпечної роботи в комп'ютерних мережах, методи створення баз даних та інтернет-ресурсів, технології розроблення алгоритмів і комп'ютерних програм мовами високого рівня із застосуванням об'єктно-орієнтованого програмування для розв'язання задач проектування і використання інформаційних систем та технологій.

126-ПР 4. Проводити системний аналіз об'єктів проектування та обґрунтовувати вибір структури, алгоритмів та способів передачі інформації в інформаційних системах та технологіях.

126-ПР 5. Аргументувати вибір програмних та технічних засобів для створення інформаційних систем та технологій на основі аналізу їх властивостей, призначення і технічних характеристик з урахуванням вимог до системи і експлуатаційних умов; мати навички налагодження та тестування програмних і технічних засобів інформаційних систем та технологій.

126-ПР 6. Демонструвати знання сучасного рівня технологій інформаційних систем, практичні навички програмування та використання прикладних і спеціалізованих комп'ютерних систем та середовищ з метою їх запровадження у професійній діяльності.

126-ПР 7. Обґрунтовувати вибір технічної структури та розробляти відповідне програмне забезпечення, що входить до складу інформаційних систем та технологій.

СИСТЕМА ОЦІНЮВАННЯ

Розподіл балів для оцінювання успішності студента	Сума балів за всі види навчальної діяльності	Оцінка ECTS	Оцінка за національною шкалою	Нарахування балів	100% підсумкове оцінювання у вигляді заліку (30%) та поточного оцінювання (70%). 30% залік 70% поточне оцінювання: Модуль №1 (10%) Модуль №2 (20%) Лабораторні роботи (40%)
	90-100	A	відмінно		
	82-89	B	добре		
	74-81	C			
	64-73	D	задовільно		
	60-63	E			
35-59	FX	незадовільно з можливістю повторного складання			

0-34

F

незадовільно з обов'язковим повторним вивченням дисципліни

Політика курсу

Студент зобов'язаний відвідувати всі заняття згідно навчального розкладу та дотримуватися норм академічної етики. Для вивчення дисципліни необхідно мати власний персональний комп'ютер та/або використовувати комп'ютери обчислювального центру кафедри. Студент повинен працювати з обов'язковою та додатковою літературою, зокрема з інформаційними ресурсами в Інтернеті. Усі лабораторні роботи мають бути виконані та здані студентом протягом семестру, у якому викладається дисципліна, до початку залікового тижня. Без особистої присутності студента підсумковий контроль не проводиться.

Структура та зміст курсу

Лекція 1	Вступ до обробки даних у Python	Лабораторна робота 1	Обробка даних та побудова графіків.	Самостійна робота	Набір статистичних інструментів Python
Лекція 2	IPython: інтерактивні обчислення та середовище розробки	Лабораторна робота 2	Створення бази даних за допомогою SQLite та робота з нею.		Робота з базами даних MySQL, PostgreSQL.
Лекція 3, 4	Основи NumPy	Лабораторна робота 3	Створення графічного інтерфейсу роботи з БД за допомогою Tkinter		Вбудовані бібліотеки для роботи з графічним інтерфейсом: створення, функціональність.
Лекція 5	Знайомство з Pandas	Лабораторна робота 4	Імпорт даних до .doc, .docx, .xls, .xlsx файлів		Побудова графіків у розробленому графічному інтерфейсі.
Лекція 6	Зчитування та запис даних, формати файлів.	Лабораторна робота 5	Створення графічного інтерфейсу роботи з БД за допомогою PyQT		Сторони бібліотеки для роботи з графікою.
Лекція 7, 8	Побудова графіків та візуалізація				
Лекція 9	Основи SQLite				
Лекція 10	Доступ до БД SQLite з Python				
Лекція 11	Знайомство з MongoDB				
Лекція 12	Програмування графічного користувацького інтерфейсу				
Лекція 13	Знайомство з PyQT5				
Лекція 14	Управління вікном додатка. Обробка сигналів та подій PyQT5.				
Лекція 15	Розміщення компонентів у вікні. Основні компоненти PyQT5				
Лекція 16	Робота з графікою PyQT5 та Tkinter				

Література

Обов'язкова	<ol style="list-style-type: none"> 1. Florent Buisson. Behavioral Data Analysis with R and Python – USA: O'Reilly, 2021 – 336p. 2. Peter Farrell. The Statistics and Calculus with Python Workshop / Peter Farrell, Alvaro Fuentes, Ajinkya Sudhir Kolhe, Quan Nguyen, Alexander Joseph Sarver, Marios Tsatsos – UK: Packt Publishing Ltd. – 705p. 3. Oliver R. Simpson. Python for Data Analysis – Independently published, 2019 – 137p 4. DR. PATRICK JEFF The advanced python for data analysis, 2020 – 60p. 5. Mehendi Hzn. Python Tricks And Tips Magazine: Gain Insider Skills : Advanced Guides & Tips – Press Publications, 2021 6. Alan D. Moore Python GUI Programming with Tkinter – Packt Publishing, 2018 – 452p 	Додаткова	<ol style="list-style-type: none"> 1. Aurelien Geron. Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow – USA: O'Reilly Media, 2019 – 856 p. 2. Jake VanderPlas Python Data Science Handbook – USA: O'Reilly Media, 2016 – 548 p. 3. Jesper Wisborg Krogh. MySQL Connector/Python Revealed – Apress, 2018 – 538p. 4. Konnor Cluster Python Machine Learning: A Step-by-Step Guide to Scikit-Learn and TensorFlow, 2019 – 126p. <p>ІНФОРМАЦІЙНІ РЕСУРСИ В ІНТЕРНЕТІ</p> <ol style="list-style-type: none"> 1. PyQt5 tutorial [Електронний ресурс]. – Режим доступу: https://build-system.fman.io/pyqt5-tutorial 2. tkinter — Python interface to Tcl/Tk [Електронний ресурс]. – Режим доступу: https://docs.python.org/3/library/tkinter.html 3. Tkinter 8.5 reference: a GUI for Python [Електронний ресурс]. – Режим доступу: https://tkdocs.com/shipman/ 4. NumPy [Електронний ресурс]. – Режим доступу: https://numpy.org/ 5. IPython Interactive Computing [Електронний ресурс]. – Режим доступу: https://ipython.org/ 6. The Jupyter Notebook [Електронний ресурс]. – Режим доступу: https://jupyter.org/ 7. Pandas [Електронний ресурс]. – Режим доступу: https://pandas.pydata.org/ 8. Matplotlib: Visualization with Python [Електронний ресурс]. – Режим доступу: https://matplotlib.org/ 9. Seaborn [Електронний ресурс]. – Режим доступу: https://seaborn.pydata.org/ 10. scikit-learn Machine Learning in Python [Електронний ресурс]. – Режим доступу: https://scikit-learn.org/stable/
Норми академічної етики			
Студент повинен дотримуватися «Кодексу етики академічних взаємовідносин та доброчесності НТУ «ХПІ»: виявляти дисциплінованість, вихованість, доброзичливість, чесність, відповідальність			
Силабус за змістом повністю відповідає робочій програмі курсу.			