

Міністерство освіти і науки України
Львівський національний університет імені Івана Франка
Кафедра радіоелектронних і комп'ютерних систем

“ЗАТВЕРДЖУЮ”

Декан факультету електроніки та
комп'ютерних технологій

Фургала Ю.М.

“ ” 2020 року

ПРОФЕСІЙНИЙ РУТНОН

ПРОГРАМА

навчальної дисципліни

підготовки бакалаврів

галузі знань 12 Інформаційні технології 126 інформаційні системи та технології

(шифр за ОПШ _____)

2020 рік

РОЗРОБЛЕНО ТА ВНЕСЕНО:

Львівським національним університетом імені Івана Франка

РОЗРОБНИКИ ПРОГРАМИ:

Сінькевич О.О., асистент кафедри радіоелектронних і комп'ютерних систем, факультету електроніки та комп'ютерних технологій

Затверджено на засіданні Вченої ради факультету електроніки та комп'ютерних технологій

“ ___ ” _____ 20 року, протокол № ___ / ___

Голова Вченої ради

Фургала Ю. М.

Обговорено та рекомендовано до затвердження Навчально-методичною радою факультету електроніки та комп'ютерних технологій

“ ___ ” _____ 20 року, протокол № ___

Голова навчально-методичної ради
факультету електроніки та комп'ютерних технологій

Лучечко А. П.

Програму схвалено на засіданні кафедри радіоелектронних і комп'ютерних систем

“ ___ ” _____ 20 року, протокол № ___

В.о. завідувача кафедри
радіоелектронних і комп'ютерних систем

Оленич І. Б.

ЗАГАЛЬНІ ВІДОМОСТІ

Програма вивчення навчальної дисципліни «Професійний Python» складена відповідно до освітньо-професійної програми підготовки здобувачів першого (бакалаврського) рівня вищої освіти спеціальності **126- інформаційні системи та технології** і є вибірковою дисципліною циклу професійно-орієнтованих дисциплін.

Міждисциплінарні зв'язки: навчальна дисципліна «Мова програмування Python» вивчається у логічному взаємозв'язку з іншими навчальними дисциплінами, що обумовлює необхідність постійного обліку та реалізації викладачами існуючих міжпредметних зв'язків з суміжними навчальними курсами, зокрема, такими як «Кіберфізичні системи», «Проектування розумних об'єктів», «Машинне навчання», «Веб-технології» та інші.

Предметом вивчення навчальної дисципліни є синтаксис, структури даних та алгоритми, модульне програмування та ООП мови програмування Python.

Програма навчальної дисципліни складається з двох модулів та дев'яти лабораторних.

1. МЕТА ТА ЗАВДАННЯ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

1.1. Метою викладання навчальної дисципліни є одержання студентами знань про синтаксис Python 3; структури даних та відповідні алгоритми; написання функцій, класів та скриптів; роботу пакетних менеджерів, віртуального середовища; під'єднання до NoSQL бази даних MongoDB; ознайомлення з середовищами розробки програм; модульним тестуванням; потоки та процеси.

1.2. Основні завдання вивчення дисципліни: навчити студентів використовувати Python для створення скриптів та реалізації основних алгоритмів; підготувати студентів до проходження алгоритмічно-програмного етапу співбесід в ІТ-компанії; дати навички для застосування отриманих знань у проектуванні алгоритмічно-орієнтованих рішень.

1.3. Згідно з вимогами освітньо-професійної програми студенти повинні:

знати: процес написання скриптів; створення добре структурованих програм на мові Python з використанням ООП; основи модульного тестування розроблених класів; типові структури даних та алгоритми з їх використанням; теоретичний та практичний матеріал згідно програми курсу: типи даних, оператори, умови, цикли, функції, об'єкти, методи і т. п.

вміти: застосовувати засвоєний матеріал для програмної реалізації алгоритмічних задач; створювати та використовувати об'єктно-орієнтований код для вирішення типових задач, а також реалізувати складні структури даних та пов'язані алгоритми.

Форма навчання	Семестр	Всього кредитів/годин	Розподіл навчального часу за видами занять					Семестр. атестація
			Лекції	Практ. зан.	Сем. зан.	Лаб. роб.	СРС	
Денна	6	6.0/180	32	-	-	32	116	залік

II. ІНФОРМАЦІЙНИЙ ОБСЯГ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Модуль 1. Об'єктно-орієнтоване програмування при роботі зі структурами даних та алгоритмами.

Тема 1.1. Мислити як computer scientist. Динамічна типізація. Класи та ООП. Особливості ООП у Python 3: інкапсуляція, наслідування, поліморфізм, композиція, абстракція. Методи класів, їх типи.

Тема 1.2. Ієрархія абстракцій. Абстрактний клас. Структура Python-проектів. ООП реалізації зв'язних списків (ЗС) (алгоритми). ЗС для об'єктів. Сортування зв'язних списків.

Тема 1.3. Масиви (списки). ООП реалізації. Контейнери об'єктів. Нерекурсивні алгоритми впорядкування об'єктів у масивах. Основи опрацювання виключень.

Тема 1.4. Словники та хеші. Робота з JSON. Серіалізація об'єктів. Основи NoSQL. Інтеграція з MongoDB. Singleton патерн. Використання логування.

Тема 1.5. Пошукові алгоритми об'єктів. Бінарний, лінійний та інтерполяційний пошуки. Основи unit-тестування.

Тема 1.6. Стеки та черги об'єктів. ООП реалізація. Циклічна черга. Стек і ЗС.

Тема 1.7. Об'єкти і рекурсія. Стек викликів. Рекурсивні алгоритми сортування об'єктів. Рекурсивний пошук. Передача аргументів між рекурсивними викликами.

Модуль 2. Складні алгоритми у прийнятті рішень. Потокі і процеси, GIL. Створення API.

Тема 2.1. Об'єкти та графи. ООП реалізації. Пошуки по графу: вглиб та вшир. Приклад API з інтеграцією Neo4j.

Тема 2.2. Дерева та об'єкти. ООП реалізації простого дерева. Бінарне дерево та пошук об'єктів. Основи B-дерева.

Тема 2.3. Дерево прийняття рішень. ООП реалізація. Основи CART алгоритму. Застосування у машинному навчанні.

Тема 2.4. Потокі та процеси у Python. Асинхронні потоки. Бібліотеки threading та multiprocessing. GIL.

Тема 2.5. MVT архітектура. Створення простого Restful API. Мікрофреймворки Flask, Fastapi. Структурування проекту.

Тема 2.6. Основи патернів проектування. Типи патернів. Декорування класів та методів.

Тема 2.7. Інтеграції коду і модулів. Система контролю версій GIT. Основи рефакторингу коду. Python - мова AI.

ІІІ. ПРИБЛИЗНИЙ ПЕРЕЛІК ЛАБОРАТОРНИХ РОБІТ

1. Об'єкти та класи. Класи шахових фігур.
2. ООП реалізація одно- та двозв'язного списку об'єктів шахової гри.
3. ООП реалізація масиву (списку) шахових фігур. Впорядкування масивів.
4. Серіалізація об'єктів та збереження JSON даних у базу даних MongoDB.
5. ООП реалізація стеку та черги об'єктів, циклічна черга.
6. ООП реалізація графа, алгоритми пошуку.
7. ООП реалізація простого дерева рішень, підготовка типів даних.
8. Створення простого Restful api з використанням Flask/Fastapi на основі розроблених шахових класів.
9. Реалізація патерна Декоратор для шахових класів. Інтеграція з базою даних MongoDB та системою Gitlab.

ІV. РЕКОМЕНДОВАНА ЛІТЕРАТУРА

1. Род Стивенс. Алгоритмы. Теория и практическое применение. Wiley.-2013. -543 с.
2. Марк Лутц. Изучаем Python, 5-е издание. Символ-Плюс, O'Reilly.-2013. -1700 с.
3. Марк Саммерфилд. Программирование на Python 3. Подробное руководство. Символ-Плюс. -2016. -608 с.
4. Brett Slatkin. Effective Python: 59 Specific Ways to Write Better Python (Effective Software Development Series). Addison-Wesley Professional; 1 edition. - 2015. - 256 с.
5. М. Гринберг. Разработка веб-приложений с использованием Flask на языке Python. ДМК Пресс. - 272 с.
6. Luciano Ramalho. Fluent Python: Clear, Concise, and Effective Programming. O'Reilly Media; 1 edition. - 2015. -792 с.
7. <https://realpython.com>

V. ФОРМА ПІДСУМКОВОГО КОНТРОЛЮ УСПІШНОСТІ НАВЧАННЯ

Контроль засвоєння матеріалу включає поточний контроль (дві модульні контрольні роботи 2x15 = 30 балів), лабораторні роботи/оцінки відповідей (60 балів) та відвідування занять (10 балів). Сумарна оцінка виставляється за 100-бальною шкалою.

Вивчення курсу завершується заліком, який проводиться в усній формі на основі семестрової успішності.

VI. ЗАСОБИ ДІАГНОСТИКИ УСПІШНОСТІ НАВЧАННЯ

Для засвоєння теоретичного матеріалу, підготовки до виконання лабораторних завдань студентам надається можливість користуватися безкоштовними онлайн-ресурсами (посилання надаються під час лекцій), бібліотеками Львівського національного університету імені Івана Франка, студентам старших курсів (починаючи з третього) – Львівською національною науковою бібліотекою України імені В. Стефаника. Також студенти можуть проходити безкоштовні онлайн-курси з дисципліни, наприклад Prometheus. Студенти мають змогу отримати консультації з питань дисципліни в лектора та викладачів, які проводять лабораторні заняття.

Оцінка якості засвоєння навчальної дисципліни включає поточний контроль успішності та складання підсумкового контролю.

Для поточного контролю засвоєння студентами навчального матеріалу передбачається виконання та захист 9 обов'язкових лабораторних робіт та написання 2 модульних контрольних робіт.

При оцінюванні лабораторної роботи враховується підготовка до виконання лабораторної роботи, хід виконання лабораторної роботи, оформлення звіту, отримані результати та захист звіту про виконану лабораторну роботу.