

Анотація навчальної дисципліни вільного вибору здобувача вищої освіти

Дисципліна:	«Скриптові мови програмування»
Викладач:	Сяський Володимир Андрійович, к.т.н., доцент
E-mail:	syasky_v@ukr.net
Кількість кредитів:	4
Мова викладання:	українська
Вид контролю:	залік
Місце у структурно-логічній схемі:	вивчається в 6 семестрі першого (бакалаврського) рівня вищої освіти за спеціальністю 122 Комп'ютерні науки

Мета та завдання дисципліни

Метою викладання дисципліни «Скриптові мови програмування» є формування у студентів знань, вмінь і навичок при вирішенні на ЕОМ різноманітних прикладних задач засобами мови програмування Python. При цьому вони повинні: знати способи структурування даних та алгоритми їх обробки; вільно володіти структурним, об'єктно-орієнтованим підходами в аналізі, проектуванні та програмуванні; вміти проводити аналіз ефективності алгоритмів і програм та проводити їх оптимізацію.

У процесі вивчення дисципліни «Скриптові мови програмування» здобувачам вищої освіти потрібно вирішити наступні **завдання**:

- ознайомитися з основними синтаксичними елементами та засобами управління процесом обчислень мови програмування Python;
- оволодіти засобами структурування даних у мові Python та алгоритмами їх обробки;
- освоїти технології програмування скриптів із використанням процедурної, функціональної і об'єктно-орієнтованої парадигм.

Зміст навчальної дисципліни

Скриптові мови програмування.

- Програмування скриптів (сценаріїв).
- Python – скриптова мова програмування.

Базові поняття мови Python.

- Базовий синтаксис мови. Лексеми. Змінні. Ввід і вивід даних.
- Систематика типів даних. Перетворення типів даних. Скалярні типи даних.
- Операції та оператори над числами. Присвоєння та переприсвоєння значень змінних.
- Логічний тип даних. Логічні оператори. Логічні вирази.

Оператори управління обчисленнями в мові Python.

- Основні алгоритмічні структури. Послідовні обчислення. Порожній оператор.
- Обчислення із розгалуженнями: умовний оператор **if**; оператор розгалуження **if-else**; оператор багатовіткового розгалуження **if-elif-else**.
- Обчислення із циклічними повтореннями: оператор циклу з передумовою **while**; оператор циклу з передумовою та альтернативою **while-else**; оператор циклу з ітератором **for**; оператори управління ітераціями циклів **break** та **continue**; вкладені цикли.

Складні структури даних мови Python.

- Рядки. Перетворення в рядки даних інших типів. Операції та оператори над рядками. Функції над рядками. Методи рядків.
- Списки. Перетворення в списки даних інших типів. Операції та оператори над списками. Функції над списками. Методи списків. Вкладені списки.

- Множини. Перетворення в множини даних інших типів. Операції та оператори над множинами. Функції над множинами. Методи множин.
- Кортежі. Перетворення в кортежі даних інших типів. Операції та оператори над кортежами. Функції над кортежами. Методи кортежів.
- Словники. Перетворення в словники даних інших типів. Операції та оператори над словниками. Функції над словниками. Методи словників.

Функціональне програмування.

- Підпрограми функцій. Параметри функцій: формальні та фактичні параметри (аргументи).
- Області видимості ідентифікаторів: глобальні та локальні змінні.
- Позиційні аргументи функцій. Іменовані аргументи функцій. Аргументи по замовчуванню. Функції з довільним набором параметрів. Вкладені функції.
- Анонімні лямбда-функції.
- Рекурсивні функції. Реалізація повторюваних обчислень рекурсивними функціями.

Модульність в Python.

- Стандартні модулі Python.
- Користувацькі модулі.
- Імпортування з модулів.

Робота з дисковими файлами.

- Відкривання файлу – створення файлового об'єкта.
- Функції та методи файлових об'єктів.
- Запис даних у файл послідовного доступу.
- Читання даних з файлу послідовного доступу.

Винятки у Python.

- Поняття винятків.
- Оброблення винятків.
- Класи вбудованих винятків.

Об'єктно-орієнтоване програмування в Python.

- Визначальний принципи ООП – інкапсуляція даних і методів під інтерфейсом.
- Оголошення класу об'єктів. Створення об'єкта як екземпляру класу. Конструктор. Вказівник «на себе» *self*.
- Побудова ієрархій класів за принципом наслідування.
- Поліморфізм методів.