

Міністерство освіти і науки України
Національний університет водного господарства та природокористування
Навчально-науковий інститут автоматики, кібернетики
та обчислювальної техніки
Кафедра прикладної математики

"ЗАТВЕРДЖУЮ"

Проректор з науково-педагогічної,
методичної та виховної роботи

_____ О.А. Лагоднюк
"___" _____ 2019 р.

04-01-52

РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ
Program of the Discipline

Сучасні та спеціалізовані мови програмування
Modern and specialized programming languages

спеціальність	113 “Прикладна математика”
specialty	113 “Applied mathematics”

Робоча програма «Сучасні та спеціалізовані мови програмування» для студентів, які навчаються за спеціальністю 113 «Прикладна математика». Рівне: НУВГП, 2019. 11 с.

Розробник:

Гаврилюк Володимир Іванович, к.т.н., доцент кафедри прикладної математики

Робочу програму схвалено на засіданні кафедри прикладної математики

Протокол від «28» серпня 2019 року № 16

Завідувач кафедри _____ П.М. Мартинюк

Схвалено науково-методичною комісією за спеціальністю 113 «Прикладна математика»

Протокол від «30» серпня_2019 року № 8

Голова науково-методичної комісії _____ П.М. Мартинюк

ВСТУП

Робоча програма нормативної навчальної дисципліни «Сучасні та спеціалізовані мови програмування» складена відповідно до освітньо-професійної програми підготовки бакалавра спеціальності «Прикладна математика».

Міждисциплінарні зв'язки: дисципліна є складовою частиною блоку вільного вибору та відноситься до навчальних дисциплін циклу загальної підготовки студентів.

Дисципліна «Сучасні та спеціалізовані мови програмування» спрямована дати студентам теоретичну та практичну підготовку з основ функціонального та об'єктно-орієнтованого програмування мовою Python.

Анотація

Дисципліна «Сучасні та спеціалізовані мови програмування» є теоретичною та практичною основою сукупності знань та вмінь, що формують профіль спеціаліста в області проектування та розробки програмного забезпечення із використанням сучасних та спеціалізованих мов програмування. В рамках вивчення даної дисципліни розглянуто сучасні парадигми програмування під час програмної реалізації прикладних завдань в області інформаційних систем і технологій та у майбутній професійній діяльності. Розглянуто особливості використання Python під час розробки різноманітних програмних продуктів.

Ключові слова: програмування, об'єктно-орієнтоване програмування, консольний додаток, веб-застосунок, веб-сервіс, Python, Flask, Django, SQLite, PyCharm.

Abstract

The discipline "Modern and Specialized Programming Languages" is a theoretical and practical basis of a set of knowledge and skills that form the profile of a specialist in the field of software design and development using modern and specialized programming languages. As part of the study of this discipline, modern programming paradigms are considered during the program implementation of applied tasks in the field of information systems and technologies and in future professional activities. Features of using Python in the development of various software products are considered

Keywords: programming, object-oriented programming, console application, web application, web service, Python, Flask, Django, SQLite, PyCharm.

1. Опис предмета навчальної дисципліни

Найменування показників	Галузь знань, спеціальність, рівень вищої освіти	Характеристика навчальної дисципліни
		денна форма навчання
Кількість кредитів – 4	Галузь знань 11 Математика та статистика	Нормативна
Модулів – 2	Спеціальність 113 Прикладна математика	Рік підготовки:
Змістових модулів – 3		2-й
Індивідуальне науково-дослідне завдання: –		Семестри
Загальна кількість годин – 120		4-й
Тижневих годин для денної форми навчання: аудиторних – 3 Самостійної роботи студента – 4,5	Рівень вищої освіти: бакалавр	Лекції
		24 год.
		Практичні, семінарські
		-
		Лабораторні
		24 год.
		Самостійна робота
72 год.		
Індивідуальні завдання:	-	
Вид контролю: іспит		

Примітка.

Співвідношення кількості годин аудиторних занять до самостійної та індивідуальної роботи становить (%):

для денної форми навчання – **40%** до **60 %**.

2. Мета та завдання навчальної дисципліни

Мета: навчальної дисципліни є вивчення основних положень мови програмування Python та застосуванні різного роду алгоритмів, бібліотек та модулів Python для створення прикладних програм.

Завдання: освоєння принципів алгоритмічного та об'єктно-орієнтованого підходів до розробки програмного забезпечення різного рівня складності.

У результаті вивчення навчальної дисципліни студент повинен

знати:

1. Принципи роботи в середовищі програмування Python ide.
2. Особливості мови програмування, синтаксис мови, основні конструкції, цикли, функції, рядки Python.
3. Основні структури даних в Python.
4. Принципи об'єктно-орієнтованого програмування мовою Python.
4. Вбудовані та додаткові модулі мови Python, які використовуються при розробці програмного забезпечення.

вміти:

1. Встановлювати і налаштовувати середовище програмування Python ide.
2. Розробляти програми різного рівня складності.
3. Використовувати вбудовані модулі Python.
4. Створювати власні модулі.
5. Проектувати і розробляти різні види програмного забезпечення на основі об'єктно-орієнтованого підходу.
6. Встановлювати та використовувати додаткові модулі мови Python при розробці програмного забезпечення.

Вимоги до знань та умінь визначаються галузевими стандартами вищої освіти України.

3. Програма навчальної дисципліни

МОДУЛЬ 1

ЗМІСТОВИЙ МОДУЛЬ 1

Основи роботи з Python.

Тема 1. Вступ до Python.

Основи мови програмування Python. Встановлення та налаштування Ide редакторів PyCharm, Visual Studio Code.

Тема 2-3. Основи Python.

Змінні та типи даних. Операції з числами. Умовні вирази. Умовні конструкції. Цикли. Функції. Область видимості змінних. Модулі. Обробка виключень. Списки. Кортежі. Словники. Множини. Операції з рядками. Основні методи рядків. Форматування рядків.

Тема 4. Робота з файлами та файловою системою.

Робота з різними типами файлів. Запис та зчитування з файлів. Модуль shelve та OS. Робота з файловою системою.

МОДУЛЬ 2 ЗМІСТОВИЙ МОДУЛЬ 2

Принципи ООП в Python та основні модулі

Тема 5. Об'єктно-орієнтоване програмування в Python.

Класи та об'єкти. Інкапсуляція. Наслідування. Поліморфізм. Клас object.

Тема 6. Основні вбудовані модулі та пакети Python та їх використання.

Модуль random, Модуль math. Модуль locale. Модуль decimal.

ЗМІСТОВИЙ МОДУЛЬ 3

Робота з базами даних, web-застосунки та сервіси.

Тема 7. Робота з базами даних в Python

Підключення і робота з базами даних SQLite, MySQL, PostgreSQL та інші в Python.

Тема 8. Скрапінг web-сайтів

Використання бібліотек: urllib, requests, BeautifulSoup.

Тема 9. Створення веб-застосунку з допомогою Flask в Python.

Flask+Redis+SQLite.

4. Структура навчальної дисципліни

Назви змістових модулів і тем	Кількість годин					
	Усього	денна форма				
		у тому числі				
		л	п	лаб.	інд.	с.р.
1	2	3	4	5	6	7
Модуль 1						
Змістовий модуль 1						
Основи роботи з Python						
Тема 1. Вступ до Python.	10	2		2		6
Тема 2-3. Основи Python.	40	8		8		24
Тема 4. Робота з файлами та файловою системою.	10	2		2		6
Разом за змістовим модулем 1	60	12		12		36
Модуль 2						
Змістовий модуль 2						
Принципи ООП в Python та основні модулі						
Тема 5. Об'єктно-орієнтоване програмування в Python.	10	2		2		6
Тема 6. Основні вбудовані модулі та пакети Python та їх використання.	14	4		4		6
Разом за змістовим модулем 2	24	6		6		12
Змістовий модуль 3						
Робота з базами даних, web-застосунки та сервіси						
Тема 7. Робота з базами даних в Python	12	2		2		8

Тема 8. Скрапінг web-сайтів	12	2		2		8
Тема 9. Створення веб-застосунку з допомогою Flask в Python.	12	2		2		8
Разом за змістовим модулем 3	36	6		6		24
Усього годин	120	24		24		72

5. Теми лабораторних занять

№ з/п	Назва теми	Кількість годин
1.	Основи роботи з Python. Встановлення, налаштування середовища розробки. Змінні та типи даних. Операції з	2
2.	Умовні конструкції. Цикли. Функції. Область видимості змінних. Модулі. Обробка виключень.	2
3.	Функції. Генератори. Обробка помилок. Виняткові ситуації.	2
4.	Списки. Кортежі. Словники. Множини.	2
5.	Операції з рядками. Основні методи рядків. Форматування рядків. Регулярні вирази.	2
6.	Робота з різними типами файлів. Запис та зчитування з файлів. Модуль shelve та OS. Робота з файловою системою	2
7.	Об'єктно-орієнтоване програмування. Основні поняття. Абстракція і декомпозиція. Об'єкти. Типи і класи. Інкапсуляція	2
8.	Відносини між класами. Спадкування. Статичний метод. Метод класу. Метакласи. Мультиметоди.	2
9.	Робота з даними в різних форматах.	2
10.	Робота з сокетами. Модуль smtplib. Модуль poplib. Функції для аналізу URL. Модулі для клієнта WWW -urllib, urllib2.	2
11.	Робота з базами даних SQLite, MySQL, PostgreSQL.	2
12.	Web фреймворк для Python.	2
	Разом	24

6. Завдання до самостійної роботи

Розподіл годин самостійної роботи для студентів включає наступні пункти:

1. Підготовка до аудиторних занять (0,5 год. на 1 год. аудиторних занять) – 24 год.
2. Підготовка до контрольних заходів (6 год. на 1 кредит) – 24 год.
3. Опрацювання окремих тем програми або їх частин, які не викладаються на лекціях – 24 год.

Розподіл навчального часу на вивчення дисципліни „Чисельні методи математичної фізики”

6.1. Розподіл самостійної роботи студента

Число кредитів ЕСТС	Загальний обсяг дисципліни	Розподіл часу		Частка самостійної роботи, в %
		Аудиторні заняття	Самостійна робота	
4	120	48	72	60

Опрацювання лекційного матеріалу	$0,5 \cdot 30 =$	14 годин
Підготовка до лабораторних робіт	$0,5 \cdot 30 =$	14 годин
Підготовка до екзамену	$6 \cdot 4 =$	24 годин
Всього		48 годин
Резерв		24 годин

6.2. Завдання для самостійної роботи

№ з/п	Назва теми	Кількість годин
1.	Засоби контролю версій (Git). Засоби обміну даними проєктів BitBucket, GitHub.	4
2.	Регулярні вирази. Синтаксис регулярного виразу.	2
3.	Підключення та налаштування бібліотек для роботи з зображеннями.	2
4.	Підключення та налаштування бібліотек для роботи з базами даних.	2
5.	Створення додатків із графічним інтерфейсом користувача.	2
6.	Мережні додатки на Python. Робота із сокетом. Модулі для клієнта WWW.	2
7.	Багатопотокові обчислення. Функції модуля threading.	2
8.	Таймер. Замки. Семафори.	4

9.	Dockerfile та docker-compose. Docker-Hub.	4
	Разом	24

Звіт про самостійну роботу не надається. Вивчення відповідного матеріалу перевіряється під час поточного та підсумкового контролів знань.

7. Методи навчання

1. Лекції проводяться з використанням технічних засобів навчання і супроводжуються демонстрацією презентацій за допомогою відеопроєктора.
2. Лабораторні роботи проводяться в комп'ютерному класі з використанням роздаткового матеріалу та методичних вказівок.
3. Проведення контрольних тестувань.

8. Методи контролю

Оцінювання навчальних досягнень студентів за усіма видами навчальних робіт проводиться за поточним та підсумковим контролями. Поточний контроль знань студентів з навчальної дисципліни проводиться за допомогою тестів. Контрольні завдання включають тестові питання трьох рівнів складності. Контроль самостійної роботи проводиться:

- з лекційного матеріалу – шляхом перевірки конспектів;
- з лабораторних робіт – за допомогою перевірки виконаних завдань та теоретичної підготовки до занять.

Усі контрольні заходи включено до 100-бальної шкали оцінювання. Підсумковий семестровий контроль знань відбувається на екзамені у формі тестування. Основними критеріями, що характеризують рівень компетентності студента при оцінюванні результатів поточного та підсумкового контролів з навчальної дисципліни «Сучасні та спеціалізовані мови програмування» є:

- виконання всіх видів навчальної роботи, що передбачені робочою програмою навчальної дисципліни;
- глибина і характер знань навчального матеріалу за змістом навчальної дисципліни;
- характер відповідей на поставлені питання (чіткість, лаконічність, логічність, послідовність тощо);
- обґрунтування вибору методу для розв'язання задач;
- рівень вміння аналізувати одержані результати.

Оцінювання результатів усіх форм контролю передбачено у 100-бальній шкалі. Критерії оцінювання результатів поточної роботи (завдань, що виконуються на практичних заняттях, результати самостійної роботи студентів) проводиться у % від кількості балів, виділених на завдання, із заокругленням до цілого числа:

0% – завдання не виконано;

40% – завдання виконано частково та містить суттєві помилки методичного або розрахункового характеру;

60% – завдання виконано повністю, але містить суттєві помилки у розрахунках або в методиці;

80% – завдання виконано повністю і вчасно, проте містить окремі несуттєві недоліки (розмірності, висновки, оформлення тощо);

100% – завдання виконано правильно, вчасно і без зауважень.

9.1. Розподіл балів, які отримують студенти

Поточне тестування та самостійна робота									Підсумковий тест (екзамен)	Сума
Змістовий модуль 1				Змістовий модуль 2		Змістовий модуль 3			40	100
30				15		15				
T1	T2	T3	T4	T5	T6	T7	T8	T9		
5	5	15	5	5	10	5	5	5		

T1, T2,...,T9 – теми змістових модулів

9.2. Шкала оцінювання

У випадку підсумкового контролю у формі заліку сума набраних балів та оцінка за 4-бальною шкалою оприлюднюються до початку екзаменаційної сесії у електронному журналі академічної групи. У екзаменаційній відомості результати навчання проставляються за двома шкалами – 100-бальною та національною. Позитивні оцінки виставляються тільки тим студентам, які виконали всі види навчальної роботи, передбачені робочою програмою навчальної дисципліни, і набрали за результатами поточного та підсумкового контролів не менше 60 балів.

Конвертація 100-бальної шкали у 4-бальну здійснюється за наступною таблицею.

Сума балів за всі види навчальної діяльності	Оцінка за національною шкалою
	для екзамену
90 – 100	відмінно
82-89	добре
74-81	
64-73	задовільно
60-63	
35-59	незадовільно з можливістю повторного складання
0-34	незадовільно з обов'язковим повторним вивченням дисципліни

10. Методичне забезпечення

Методичне забезпечення навчальної дисципліни «Сучасні та спеціалізовані мови програмування» включає:

1. Опорний конспект лекцій (електронний і паперовий варіанти) за всіма темами курсу.
2. Пакети тестових завдань для всього курсу дисципліни (навчальна платформа Moodle).

11. Рекомендована література

Базова

1. Бэрри Пол. Изучаем программирование на Python. Пер. с англ. М. А. Райтман. Москва: Издательство «Э», 2017. 624 с.
2. Васильев О.М. Програмування мовою Python. Тернопіль: Навчальна книга – Богдан, 2019. 504 с.
3. Головатый А., Каплан-Мосс Дж. Django. Подробное руководство, 2-е издание. Пер. с англ. СПб.: Символ-Плюс, 2010. 560 с.
4. Дронов В. А. Django: Практика создания Webсайтов на Python. СПб.: БХВ-Петербург, 2016. 528 с.
5. Лутц М. Программирование на Python, том I, 4-е издание. Пер. с англ. СПб.: Символ-Плюс, 2011. 992 с.
6. Лутц М. Программирование на Python, том II, 4-е издание. Пер. с англ. СПб.: Символ-Плюс, 2011. 992 с.

Допоміжна

7. Прохоренко Н.А., Дронов В.А. Python 3 и PyQt 5. Разработка приложений. СПб.: БХВ-Петербург, 2016. 832 с.
8. Саммерфилд М. Программирование на Python 3. Подробное руководство. Пер. с англ. СПб.: Символ-Плюс, 2009. 608 с.
9. Форсье Дж., Биссекс П., Чан У. Django. Разработка веб-приложений на Python. Пер. с англ. СПб.: Символ-Плюс, 2010, 456 с.
10. Митчелл Р. Скрапинг веб-сайтов с помощью Python / пер. с англ. А. В. Груздев. М.: ДМК Пресс, 2016. 280 с.
11. Daniel Rubio. Beginning Django: Web Application Development and Deployment with Python. Apress. 2017. (<https://doi.org/10.1007/978-1-4842-2787-9>).

12. Інформаційні ресурси

1. Рівненська обласна універсальна бібліотека. URL: <http://libr.rv.ua/>
2. Наукова бібліотека НУВГП. URL: <http://lib.nuwm.edu.ua/>
3. Національна бібліотека ім. В.І. Вернадського URL: <http://www.nbuv.gov.ua/>