



Програмування і алгоритмічні мови

Робоча програма навчальної дисципліни (Силабус)

Реквізити навчальної дисципліни

Рівень вищої освіти	<i>Перший (бакалаврський)</i>
Галузь знань	<i>15 Автоматизація та приладобудування¹</i>
Спеціальність	<i>153 Мікро- та наносистемна техніка</i>
Освітня програма	<i>Мікро- та наноелектроніка (для бакалаврів і магістрів), Мікро- та наносистемна техніка (для PhD)</i>
Статус дисципліни	<i>Нормативна</i>
Форма навчання	<i>очна(денна)</i>
Рік підготовки, семестр	<i>2 курс, весняний семестр</i>
Обсяг дисципліни	<i>5 кредитів (150 годин)</i>
Семестровий контроль/ контрольні заходи	<i>залік</i>
Розклад занять	
Мова викладання	<i>Українська</i>
Інформація про керівника курсу / викладачів	Лектор: <i>д.т.н., доц., Татарчук Д.Д., dmitry.tatarchuk@gmail.com, м. 0971521861</i> Лабораторні: <i>к.т.н., доц., Діденко Ю.В., yu.didenko@kpi.ua, м. 0967521754</i>
Розміщення курсу	https://meet.google.com/lookup/aazexbp3ud?authuser=0&hs=179
Код курсу	<i>сuhcd5x</i>

¹ В полях Галузь знань/Спеціальність/Освітня програма:

Для дисциплін професійно-практичної підготовки зазначається інформація відповідно до навчального плану.
Для соціально-гуманітарних дисциплін вказується перелік галузей, спеціальностей, або «для всіх».

1. Опис навчальної дисципліни, її мета, предмет вивчення та результати навчання

Мета навчальної дисципліни формування у студентів здатностей:

- організації автоматизованого робочого місця на базі персональної ЕОМ;
- використання засобів обчислювальної техніки та програмного забезпечення у професійній діяльності;
- кваліфікованого використання програмного забезпечення загального призначення;
- кваліфікованого використання програмного забезпечення професійного призначення;
- розробки програмного забезпечення для розв'язку типових задач професійного спрямування з використанням сучасних технологій.

Основні завдання навчальної дисципліни:

- дати знання з основ розробки прикладного програмного забезпечення сучасними програмними засобами;
- виробити вміння розробляти прикладне програмне забезпечення для розв'язку інженерних завдань професійної спрямованості;
- сформувати досвід практичного використання апаратних та програмних засобів обчислювальної техніки, програмування з використанням сучасних засобів та технологій.

Пререквізити та постреквізити дисципліни (місце в структурно-логічній схемі навчання за відповідною освітньою програмою)

Дисципліна базується на дисципліні «Інформатика-1» і забезпечує підготовку до вивчення курсів «Обчислювальна математика», «Моделювання технологій напівпровідникових матеріалів та інтегральних мікросхем» і кредитного модуля «Моделювання в електроніці». Крім того знання, набуті в результаті вивчення даного курсу, використовуються при підготовці дипломних проектів та робіт, а також у курсовому проектуванні.

Зміст навчальної дисципліни

Вступ до курсу «Програмування і алгоритмічні мови»

Тема 1. Основні поняття теорії об'єктно орієнтованого програмування

Тема 2 Елементи мови Python

Тема 3 Функції і модулі у мові Python

Тема 4 Типи даних у мові Python

Тема 5 Обробка виняткових ситуацій у мові Python

Тема 6 Операції введення-виведення у мові Python

Тема 7 ООП у мові Python

Тема 8 Графічні можливості мови Python

Тема 9 Потоки і процеси у мові Python

Тема 10 Мережеве програмування мовою Python

2. Навчальні матеріали та ресурси

Базова література:

1. Мусин Д. Самоучитель Python [текст] / Д. Мусин. , 2017. – 154 с.
2. Федоров, Д. Ю. Программирование на языке высокого уровня Python : учеб. пособие для прикладного бакалавриата / Д. Ю. Федоров. — 2-е изд., перераб. и доп. — М. : Издательство Юрайт, 2019. — 161 с.
3. Мельник І.В. Основи програмування на мові Python. Том 1. Базові принципи побудови мови програмування Python та головні синтаксичні конструкції: комплексний навч. Посібник з курсів «Об'єктно-орієнтоване програмування» та «обчислювальні системи та мережі» для студентів-бакалаврів, які навчаються за освітньою програмою «Електронні прилади та пристрої» - К., Кафедра, 2020. – 372 с. ISBN 978-617-7301-71-3 (заг.) ISBN 978-617-7301-72-0 (Кн. 1)
Додаткові матеріали та ресурси:

1. Сайт кафедри мікроелектроніки. Розділ електронна бібліотека:
<http://me.kpi.ua/index.php?id=61>
2. <https://tproger.ru/>
3. <https://younglinux.info/>
4. <https://pythoner.name/documentation/library/functions>
7. <https://pythonz.net/references/named/range>
8. <https://pythonworld.ru/>
9. <https://pythonru.com/primery/primery-primeneniya-regulyarnyh-vyrazheniy-v-python>
10. https://pyneng.readthedocs.io/ru/latest/book/15_module_re/
11. <https://python-scripts.com/>
12. <https://www.internet-technologies.ru/articles/funkcii-print-v-python.html>
13. <http://pythonicway.com/python-fileio>
14. https://ru.wikibooks.org/wiki/GUI_Help/Tkinter_book
15. https://www.ibm.com/developerworks/ru/library/l-python_part_9/index.html
16. <https://dev-gang.ru/article/mnogopotocznost-v-python-t2bkyunvku/>
17. <https://habr.com/ru/post/149420/>
18. <https://docs.python.org/>
19. <https://coderlessons.com/>
20. <https://zametkinapolyah.ru/servera-i-protokoly/o-modeli-vzaimodejstviya-klient-server-prostymi-slovami-arhitektura-klient-server-s-primerami.html>
21. <https://echo.lviv.ua/dev/6455>

Навчальний контент

3. Методика опанування навчальної дисципліни (освітнього компонента)

Лекційні заняття:

Лекція №1

Вступ до курсу «Об'єктно-орієнтоване програмування»

Тема 1. Основні поняття теорії об'єктно орієнтованого програмуванняю

Лекція №2

Тема 1. Основні поняття теорії об'єктно орієнтованого програмуванняю (продовження)

Лекція №3

Тема 2 Елементи мови Python

Лекція №4

Тема 2 Елементи мови Python (продовження)

Лекція №5

Тема 3 Функції і модулі у мові Python

Лекція №6

Тема 3 Функції і модулі у мові Python (продовження)

Лекція №7

Тема 3 Функції і модулі у мові Python (продовження)

Лекція №8

Тема 4 Типи даних у мові Python

Лекція №9

Тема 4 Типи даних у мові Python (продовження)

Лекція №10

Тема 4 Типи даних у мові Python (продовження)

Лекція №11

Тема 5 Обробка виняткових ситуацій у мові Python

Лекція №12

Тема 6 Операції введення-виведення у мові Python

Лекція №13

Тема 7 ООП у мові Python

Лекція №14

Тема 8 Графічні можливості мови Python

Лекція №15

Тема 8 Графічні можливості мови Python (продовження)

Лекція №16

Тема 9 Потоки і процеси у мові Python

Лекція №17

Тема 9 Потоки і процеси у мові Python (продовження)

Лекція №18

Тема 10 Мережеве програмування мовою Python

Лабораторні заняття:

Заняття №1

Установка і налагодження засобів розробки програмного забезпечення мовою програмування Python

Заняття №2

Інформаційні ресурси для вивчення мови програмування Python

Заняття №3

Створення файлів програми та робота з ними

Заняття №4

Знайомство з системами програмування на мові програмування Python. Засоби налагодження програм в системах програмування на мові програмування Python.

Заняття №5

Знайомство з системами програмування на мові програмування Python. Засоби налагодження програм в системах програмування на мові програмування Python. (продовження)

Заняття №6

Оператори вводу-виводу в мові програмування Python

Заняття №7

Оператори вводу-виводу в мові програмування Python (продовження)

Заняття №8

Оператори циклу мови програмування Python

Заняття №9

Оператори циклу мови програмування Python (продовження)

Заняття №10

Оператори вибору в мові програмування Python

Заняття №11

Оператори вибору в мові програмування Python (продовження)

Заняття №12

Одновимірні масиви в мові програмування Python

Заняття №13

Одновимірні масиви в мові програмування Python (продовження)

Заняття №14

Двовимірні масиви в мові програмування Python

Заняття №15

Двовимірні масиви в мові програмування Python (продовження)

Заняття №16

Обробка символічних та строкових даних в мові програмування Python

Заняття №17

Обробка символічних та строкових даних в мові програмування Python (продовження)

Заняття №18

Робота з файлами в мові програмування Python

Заняття №19

Робота з файлами в мові програмування Python (продовження)

Заняття №20

Графічні можливості мови програмування Python

Заняття №21

Графічні можливості мови програмування Python (продовження)

Заняття №22

Приймання лабораторних робіт

Заняття №23

Приймання лабораторних робіт

Заняття №24

Приймання розрахункової роботи

Заняття №25

Приймання розрахункової роботи

Заняття №26

Підготовка до заліку

Заняття №27

Залік

4. Самостійна робота студента/аспіранта

Розрахункова робота

Для стимуляції самостійної роботи студентів, заохочення їх до самовдосконалення та знайомства з новітніми інформаційними технологіями в кредитному модулі передбачено в якості індивідуального завдання розрахункова робота за тематикою кредитного модуля. В роботі необхідно засобами мови Python реалізувати за індивідуальним завданням просту базу даних з реалізацією функцій створення бази даних, додавання записів, вилучення записів, пошуку інформації за заданим критерієм. На виконання завдання передбачено 4 години самостійної роботи.

Також на самостійну роботу додатково винесено вивчення наступного теоретичного матеріалу:

Повторне використання коду – 2 години;

Конструктор за замовчуванням -2 години;

Доступ до полів та методів предків – 2 години;

Класи друзі класів – 8 годин;

Файлові потоки – 10 годин;

Графічні можливості мови Python – 6 години;

Мережеве програмування мовою Python – 6 годин;

Підготовка до заліку – 20 годин.

Політика та контроль

5. Політика навчальної дисципліни (освітнього компонента)

Відвідування всіх видів занять є обов'язковим.

Виконання всіх завдань є обов'язковою умовою допуску до заліку.

Перед лабораторними заняттями необхідно попередньо ознайомитись з завданням на дане заняття. Завдання повинне бути виконане і показане викладачеві не пізніше заняття за наступною темою. В противному випадку знімаються рейтингові бали згідно вимог PCO.

Захист лабораторних робіт може здійснюватися на протязі всього семестру. На захист повинен бути оформлений протокол виконаної роботи. Протокол оформлюється за

загальними правилами університету і повинен містити титульний аркуш, завдання і код програми. Процедура захисту складається з відповідей на запитання викладача за темою роботи. За неправильні відповіді або неправильне оформлення протоколу оцінка знижується згідно вимог PCO. За умови неправильної відповіді більш ніж на третину запитань захист не зараховується.

Розрахункова робота також повинна бути захищена. На захист повинна бути пред'явлена виконана робота в електронному та друкованому варіантах. Друкований варіант повинен бути оформлений згідно вимог університету і повинен містити титульний аркуш, завдання, програмний код, опис всіх змінних, функцій і об'єктів, які використано у коді. Процедура захисту складається з відповідей на запитання викладача за темою роботи. За неправильні відповіді або неправильне оформлення роботи оцінка знижується згідно вимог PCO. За умови неправильної відповіді більш ніж на третину запитань захист не зараховується.

6. Види контролю та рейтингова система оцінювання результатів навчання (PCO)

З метою контролю процесу засвоєння учбового матеріалу у курсі передбачено дві модульні контрольні роботи. Перша робота за матеріалом лекцій 1...4, друга за матеріалами лекцій 5-12. Оцінювання контрольних робіт здійснюється згідно рейтингової системи. За неправильні відповіді бали не зараховуються, за неточні або не повні відповіді бали знижуються.

Виконання лабораторних робіт та розрахункової роботи проводиться шляхом перевірки функціонування написаних студентом програм та шляхом опитування в процесі захисту роботи.

Календарний контроль: провадиться двічі на семестр як моніторинг поточного стану виконання вимог силабусу. Для успішного проходження першого календарного контролю: студент повинен набрати не менше 20% балів від максимального сумарного рейтингу протягом семестру. Для успішного проходження другого календарного контролю студент повинен набрати не менше 40% балів від максимального рейтингу.

Семестровий контроль здійснюється у вигляді заліку.

Студенти, які набрали протягом семестру необхідну для позитивної оцінки кількість балів мають можливість:

- не складати залік, а отримати оцінку „ автоматом ” відповідно до набраного рейтингу з дисципліни;

- складати залік з метою підвищення оцінки.

У разі отримання на заліку оцінки нижчої, ніж за рейтингом, за студентом не зберігається оцінка отримана „ автоматом ”.

Студенти, семестровий рейтинг яких відповідає оцінці „ незадовільно ”, зобов'язані складати залік.

Студенти, які за семестровим рейтингом не допущені до заліку з цієї дисципліни, зобов'язані підвищити його до рівня не менше 60%.

Оцінка визначається за сумою набраних рейтингових балів відповідно до системи розрахунку шкали рейтингу.

Рейтинговий бал студента нараховується за наступними правилами:

1. Своєчасне виконання оцінюваних лабораторних робіт

Ваговий бал 1. Максимальна кількість балів $1 \times 8 = 8$. Бали нараховуються в разі своєчасного правильного виконання завдань передбачених лабораторними роботами. За не своєчасне виконання завдань бали не нараховуються.

2. Захист лабораторних робіт

Ваговий бал 3. Максимальна кількість балів $3 \times 8 = 24$. Бали нараховуються за результатами захисту робіт. Захист полягає у відповіді на 3-запитання викладача. За правильну відповідь на запитання нараховується 1 бал. За неправильну відповідь бали не нараховуються. Якщо не зараховано два або більше запитань захист не зараховується. В цьому випадку студент повинен підготуватися і прийти на повторний захист.

3. Модульна контрольна робота 1 – максимальний бал 20.
4. Модульна контрольна робота 2 – максимальний бал 20.
5. РГР – максимальний бал 28.

Система рейтингових (вагових) балів

№ п/п	Заняття, що підлягають рейтинговій оцінці	Загальна кількість	Макс. бал	Число балів на відмінно
1.	Оцінювані лабораторні роботи:			
	своєчасне виконання	8	1	8
	захист	8	3	24
2.	Модульна контрольна робота 1	1	20	20
3.	Модульна контрольна робота 2	1	20	20
4.	РГР	1	28	28
5.	Рейтинг за курс, R			100

Таблиця відповідності рейтингових балів оцінкам за університетською шкалою:

Кількість балів	Оцінка
100-95	Відмінно
94-85	Дуже добре
84-75	Добре
74-65	Задовільно
64-60	Достатньо
Менше 60	Незадовільно
Не виконані умови допуску	Не допущено

7. Додаткова інформація з дисципліни (освітнього компонента)

Перелік запитань для семестрового контролю.

1. З яких елементів складається мова програмування?
2. Що таке коментар? Як оформлюють коментарі у мові програмування Python? Наведіть приклад.
3. Що таке константа? Які константи можна використовувати у мові програмування Python? Наведіть приклади.
4. Що таке ідентифікатор? Які правила формування ідентифікаторів у мові програмування Python? Наведіть приклади правильних і неправильних ідентифікаторів.
5. Що таке зарезервовані слова? Наведіть приклади зарезервованих слів мови програмування Python.
6. Що таке операції і операнди? Які групи операцій допустимі у мові програмування Python?
7. Перерахуйте і опишіть арифметичні операції мови програмування Python.
8. Перерахуйте і опишіть операції відношення мови програмування Python.
9. Перерахуйте і опишіть булевські операції мови програмування Python. Наведіть таблиці істинності булевських операцій.
10. Перерахуйте і опишіть бітові операції мови програмування Python. Наведіть таблиці істинності бітових операцій.
11. Опишіть порядок пріоритетності операцій мови програмування Python.
12. Що таке спеціальні символи? Перерахуйте і опишіть спеціальні символи мови програмування Python.

13. Що таке умовний оператор? Який синтаксис умовного оператора мови програмування Python? Наведіть приклади.
14. Який синтаксис оператора циклу `while` мови програмування Python? Наведіть приклади.
15. Який синтаксис оператора покрокового циклу мови програмування Python? Наведіть приклади.
16. Який синтаксис оператора `break`? Для чого він використовується? Наведіть приклади.
17. Який синтаксис оператора `continue`? Для чого він використовується? Наведіть приклади.
18. Який синтаксис оператора присвоювання? Для чого він використовується? Наведіть приклади.
19. Перерахуйте складні оператори мови програмування Python. Наведіть приклади.
20. Для чого використовують оператор `return`? Який синтаксис оператора `return`? Наведіть приклади.
21. Дайте визначення функції. Наведіть приклади.
22. Що таке анонімна функція? Який синтаксис анонімною функції?
23. Як визначити функцію? Наведіть приклади.
24. Яким чином можна використати ім'я функції у якості аргумента іншої функції? Наведіть приклад.
25. Поясніть, що таке аргументи за замовчуванням. Наведіть приклади.
26. Поясніть, що таке іменовані аргументи. Як їх використовують? Наведіть приклади.
27. Що таке вбудовані функції? Наведіть приклади.
28. Що таке локальні змінні? Чим вони відрізняються від глобальних? Наведіть приклади.
29. Для чого використовують ключове слово `global`? Наведіть приклади.
30. Що таке нелокальна змінна? Наведіть приклади.
31. Що таке модуль? Для чого їх використовують? Наведіть приклади.
32. Як підключити до програми модуль? Наведіть приклади.
33. Опишіть бібліотечний модуль `math`.
34. Опишіть бібліотечний модуль `cmath`.
35. Опишіть бібліотечний модуль `string`.
36. Опишіть бібліотечний модуль `sys`.
37. Опишіть бібліотечний модуль `time`.
38. Опишіть бібліотечний модуль `random`.
39. Опишіть бібліотечний модуль `re`.
40. Перерахуйте основні вбудовані типи мови Python.
41. Опишіть тип `int` мови Python.
42. Опишіть тип `float` мови Python.
43. Опишіть тип `complex` мови Python.
44. Опишіть булевський тип мови Python.
45. Опишіть рядковий тип мови Python.
46. Опишіть тип список мови Python.

47. *Опишіть тип кортеж мови Python.*
48. *Опишіть тип діапазон мови Python.*
49. *Опишіть тип множина мови Python.*
50. *Опишіть тип словник мови Python.*
51. *Опишіть тип bytes мови Python.*
52. *Опишіть тип bytearray мови Python.*
53. *Опишіть тип memoryview мови Python.*
54. *Яким чином у Python обробляють помилки і виняткові ситуації? Наведіть приклади.*
55. *Опишіть синтаксис оператора try. Наведіть приклади.*
56. *Наведіть кілька прикладів виняткових ситуацій. В яких випадках вони виникають?*
57. *Який синтаксис функції print? Як її використовують? Наведіть приклади.*
58. *Який синтаксис функції input? Як її використовують? Наведіть приклади.*
59. *Опишіть клас файл. Як його використовують? Наведіть приклади.*
60. *Яка основна ідея ООП?*
61. *Що таке клас?*
62. *Які основні принципи ООП?*
63. *Поясніть, що означає термін абстрагування.*
64. *Поясніть, що означає термін інкапсуляція.*
65. *Поясніть, що означає термін успадкування.*
66. *Поясніть, що означає термін поліморфізм.*
67. *Поясніть, що означає термін об'єкт.*
68. *Поясніть, що означає термін атрибуту класу.*
69. *Як створити клас користувача у мові програмування Python?*
70. *Для чого потрібен покажчик self?*
71. *Поясніть термін композиція.*
72. *Чим композиція відрізняється від успадкування?*
73. *Для чого використовують метод __init__?*
74. *Яким чином реалізується поліморфізм у мові Python?*
75. *Яким чином реалізовано у мові Python перенавантаження функцій?*
76. *Що таке декоратори? Для чого їх використовують?*
77. *Що таке графічний інтерфейс користувача? Яким чином він реалізований у мові програмування Python?*
78. *Які дії необхідно виконати для реалізації програми з графічним інтерфейсом користувача?*
79. *Опишіть клас Tk.*
80. *Опишіть віджет Button.*
81. *Опишіть віджет Radiobutton.*
82. *Опишіть віджет Checkbutton.*

83. *Опишіть віджет Entry.*
84. *Опишіть віджет Text.*
85. *Опишіть віджет Label.*
86. *Опишіть віджет Scale.*
87. *Опишіть віджет Scrollbar.*
88. *Опишіть віджет Frame.*
89. *Опишіть віджет LabelFrame.*
90. *Опишіть віджет Listbox.*
91. *Опишіть віджет Canvas.*
93. *Поясніть, що таке потік у програмуванні.*
94. *Перерахуйте переваги і недоліки багатопотокового програмування.*
95. *В чому полягає основний недолік реалізації багатопоточності у Python?*
96. *Опишіть клас Thread.*
97. *Опишіть клас Lock.*
98. *Що таке процес?*
99. *Які переваги дає використання кількох процесів?*
100. *Який модуль містить методи для роботи з процесами?*
101. *Які кроки необхідно виконати для породження і запуску процесу?*
102. *Які режими виконання процесів підтримує мова програмування Python?*
103. *Що таке фоновий процес?*
104. *Як можна програмно завершити дочірній процес?*
105. *Які способи обміну даними між процесами розглянуто в нашому курсі?*
106. *Який принцип дії черги?*
107. *Яким чином працює канал?*
108. *Що означає термін технологія клієнт-сервер?*
109. *Які переваги технології клієнт-сервер?*
110. *Які недоліки технології клієнт-сервер?*
111. *Які функції виконує програма сервер?*
112. *Які функції виконує програма клієнт?*
113. *Що таке сокет?*
114. *Для чого використовують IP-адресу?*
115. *Для чого використовують номер порту?*

За наявності сертифікатів проходження дистанційних чи онлайн курсів за відповідною тематикою можливе зарахування курсу повністю або частково за результатами співбесіди викладача зі студентом за темою курсу.

Робочу програму навчальної дисципліни (силабус):

Складено доцентом кафедри мікроелектроніки, к.т.н., доц., Татарчуком Д.Д.

Ухвалено кафедрою _____ (протокол № ___ від _____)

Погоджено Методичною комісією факультету² (протокол № __ від _____)

² Методичною радою університету – для загальноуніверситетських дисциплін.