СИЛАБУС НАВЧАЛЬНОЇ ДИСЦИПЛІНИ
«ПРОГРАМУВАННЯ»

Компонент освітньої програми –
обов'язковий (6,5 *кредитів*)

Освітньо-професійна програма	Технології програмування та комп'ютерне моделювання
Спеціальність	F1 Прикладна математика
Галузь знань	F Інформаційні технології
Рівень вищої освіти	перший бакалаврський
Мова навчання	українська
Профайл викладача (-ів)	Скутар Ігор Дмитрович, канд. фіз.-мат. наук, асистент кафедри ПМІТ *https://amit.chnu.edu.ua/pro-kafedru/spivrobitnyky/skutar-ihor-dmytrovych/*
Контактний тел.	+38(050)9699747
E-mail:	i.skutar@chnu.edu.ua
Сторінка курсу в Moodle	*https://moodle.chnu.edu.ua/course/view.php?id=4423*
Консультації	Вівторок, 14:30 – 15:30

АНОТАЦІЯ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Дисципліна «Програмування» є обов'язковою дисципліною зі спеціальності F1 «Прикладна математика» для освітньої програми «Технології програмування та комп'ютерне моделювання», яка викладається в І семестрі в обсязі 6,5 кредитів (за Європейською Кредитно-Трансферною Системою ECTS). Дисципліна «Програмування» є розпочинає курс вивчення спеціальних дисциплін, здобуті знання знадобляться в подальшій практичній діяльності майбутнім фахівцям галузі ІТ. Курс спрямований на формування теоретичної бази знань студентів щодо побудови алгоритмів та їх реалізації мовою програмування а також розвиток уміння розв'язувати алгоритмічні задачі, користуючись відомими теоретичними положеннями, математичним апаратом, літературою та комп'ютерною технікою.

Мета навчальної дисципліни: забезпечення студентів необхідними теоретичними знаннями і практичними навиками використання мови програмування python для побудови алгоритмів та їх реалізації мовою програмування.

НАВЧАЛЬНИЙ КОНТЕНТ ОСВІТНЬОЇ КОМПОНЕНТИ

МОДУЛЬ 1.	
Тема 1	**ОСНОВНІ ПОНЯТТЯ АЛГОРИТМІЗАЦІЇ ТА ПРОГРАМУВАННЯ.**
Тема 2	**МОВА ПРОГРАМУВАННЯ PYTHON.**
Тема 3	**ЛОГІЧНІ ОПЕРАТОРИ І ВИРАЗИ.**
Тема 4	**ЦИКЛІЧНІ КОНСТРУКЦІЇ.**
Тема 5	**ОДНОВИМІРНІ МАСИВИ. НАЙПРОСТІШІ АЛГОРИТМИ РОБОТИ З**

	ОДНОВИМІРНИМИ МАСИВАМИ.
Тема 6	**УПОРЯДКУВАННЯ МАСИВІВ.**
Тема 7	**ДВОВИМІРНІ МАСИВИ.**
Тема 8	**БАЗОВІ АЛГОРИТМИ ДЛЯ ОБРОБКИ ЕЛЕМЕНТІВ ДВОВИМІРНИХ МАСИВІВ.**
Тема 9	**СЛОВНИКИ.**
МОДУЛЬ 2.	
Тема 10	**РЯДКИ І СИМВОЛЬНІ МАСИВИ. СТАНДАРТНІ ФУНКЦІЇ ДЛЯ РОБОТИ З РЯДКАМИ.**
Тема 11	**НАЙПРОСТІШІ АЛГОРИТМИ РОБОТИ ІЗ СИМВОЛЬНИМИ ТА РЯДКОВИМИ ВЕЛИЧИНАМИ.**
Тема 12	**ФУНКЦІЇ КОРИСТУВАЧА.**
Тема 13	**МОДУЛІ.**
Тема 14	**РОБОТА ІЗ ФАЙЛАМИ.**
Тема 15	**МАТЕМАТИЧНІ ПАКЕТИ.**

ФОРМИ, МЕТОДИ ТА ОСВІТНІ ТЕХНОЛОГІЇ НАВЧАННЯ

1. Форми навчання:

- **Лекції:** на лекційних заняттях студентам надається теоретичний матеріал з алгоритмізації та основ мови програмування Python, його синтаксису, стандартних бібліотек та інструментів. Лекційні заняття слугують основою для знайомства з основними концепціями програмування, такими як змінні, типи даних, оператори, цикли та умовні конструкції. Лекції можуть бути проведені як в аудиторному форматі, так і дистанційно з використанням відеоконференц-зв'язку.
- **Практичні заняття:** практика допомагає студентам застосовувати теоретичні знання на практиці. Студенти виконують базові завдання: написання простих програм, робота з функціями, створення алгоритмів.
- **Лабораторні роботи:** лабораторні заняття зосереджені на виконанні комплексних завдань, які включають елементи проєктування алгоритмів, налагодження та тестування програмного коду. Це допомагає розвинути навички аналізу та вирішення задач.
- **Індивідуальна робота:** виконання самостійних завдань та підготовка проєктів або досліджень. Це може включати як вивчення додаткової літератури, так і реалізацію власних міні-проєктів на Python.

2. Методи навчання:

- **Пояснювально-ілюстративний метод:** викладач пояснює теоретичний матеріал, супроводжуючи пояснення прикладами коду та наочними ілюстраціями. Це допомагає студентам краще зрозуміти абстрактні концепції.
- **Репродуктивний метод:** студенти повторюють типові задачі та алгоритми, які розглядаються на заняттях. Це дозволяє їм закріпити базові навички програмування.
- **Проблемний метод:** викладач пропонує задачі, які потребують творчого підходу та аналізу. Студенти вчаться шукати оптимальні рішення, експериментуючи з різними підходами.
- **Метод проєктів:** студенти працюють над створенням власного проєкту на

Python, що може включати розробку веб-додатку, автоматизацію процесів або аналіз даних.

3. Освітні технології:

- **Використання інтерактивних середовищ розробки:** середовище PyCharm активно використовуються під час навчання для написання коду, його тестування та відлагодження.
- **Системи управління навчанням (LMS):** використовується система Moodle для організації дистанційного навчання, завантаження матеріалів та комунікації зі студентами.

Такий підхід до викладання дисципліни "Програмування" дозволяє студентам, які тільки починають вивчати цю сферу, розвинути фундаментальні навички програмування та зрозуміти основні принципи розробки програмного забезпечення.

ФОРМИ Й МЕТОДИ КОНТРОЛЮ ТА ОЦІНЮВАННЯ

***Поточний контроль*:** Контрольне тестування після модуля 1, практична контрольна робота, вибіркове опитування на лекційних заняттях, виконання і захист (заліки) лабораторних робіт 1-8.

Підсумковий контроль *– залік.*

КРИТЕРІЇ ОЦІНЮВАННЯ РЕЗУЛЬТАТІВ НАВЧАННЯ

Оцінювання програмних результатів навчання здобувачів освіти здійснюється за шкалою європейської кредитно-трансферної системи (ECTS).

Критерієм успішного оцінювання є досягнення здобувачем вищої освіти мінімальних порогових рівнів (балів) за кожним запланованим результатом навчання.

ПОЛІТИКА АКАДЕМІЧНОЇ ДОБРОЧЕСНОСТІ

Дотримання політики щодо академічної доброчесності учасниками освітнього процесу при вивченні освітнього компонента регламентовано чинними документами ЧНУ:

- «Кодекс академічної доброчесності та етики Чернівецького національного університету імені Юрія Федьковича» https://www.chnu.edu.ua/universytet/normatyvni-dokumenty/kodeks-akademichnoi-dobrochesnosti-ta-etyky-chernivetskoho-natsionalnoho-universytetu-imeni-yuriia-fedkovycha/
- «Положення про виявлення та запобігання академічного плагіату у Чернівецькому національному університеті імені Юрія Федьковича» https://www.chnu.edu.ua/universytet/normatyvni-dokumenty/polozhennia-pro-vyiavlennia-ta-zapobihannia-akademichnomu-plahiatu-u-chernivetskomu-natsionalnomu-universyteti-imeni-yuriia-fedkovycha/
- «Політика використання штучного інтелекту в Чернівецькому національному університету імені Юрія Федьковича» https://www.chnu.edu.ua/universytet/normatyvni-dokumenty/polityka-vykorystannia-shtuchnoho-intelektu-v-chernivetskomu-natsionalnomu-universyteti-imeni-yuriia-fedkovycha/

ВИКОРИСТАННЯ ЗДОБУВАЧАМИ ОСВІТИ ІНСТРУМЕНТІВ ШТУЧНОГО ІНТЕЛЕКТУ (ШІ) ПІД ЧАС ВИКОНАННЯ НАВЧАЛЬНИХ ЗАВДАНЬ:

- *допускається за умови* дотримання принципів академічної доброчесності (здобувач освіти несе відповідальність за зміст, достовірність і результати поданої роботи, а у разі використання ШІ зобов'язаний зазначити факт і спосіб його застосування);
- *не допускається* використання ШІ як заміни самостійного виконання завдань або з метою порушення вимог академічної доброчесності;
- *додаткові обмеження чи вимоги* щодо використання ШІ можуть визначатися викладачем в освітньому процесі для конкретних видів навчальної діяльності відповідно до особливостей дисципліни.

ІНФОРМАЦІЙНІ РЕСУРСИ

1. Python Software Foundation. Python Official Website [Електронний ресурс]. – Режим доступу: https://www.python.org/.
2. Python Software Foundation. Python 3 Documentation [Електронний ресурс]. – Режим доступу: https://docs.python.org/3/.
3. Python Software Foundation. Python Tutorial [Електронний ресурс]. – Режим доступу: https://docs.python.org/3/tutorial/.
4. Python Tutor. Visualize Python, Java, C, C++, and JavaScript Code Execution [Електронний ресурс]. – Режим доступу: https://pythontutor.com/.
5. Подоба В. Python 3: Курс молодого бійця [Електронний ресурс]. – Режим доступу: http://www.vitaliypodoba.com/tutorials/python3-beginners-course/.

Детальна інформація щодо вивчення курсу «Назва навчальної дисципліни» висвітлена у робочій програмі навчальної дисципліни:

https://media.chnu.edu.ua/media/ug5m2yey/f1_ok6_rp_programming_python.pdf