



**ЗАХІДНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ
УНІВЕРСИТЕТ
ФАКУЛЬТЕТ КОМП'ЮТЕРНИХ ІНФОРМАЦІЙНИХ
ТЕХНОЛОГІЙ
КАФЕДРА ЕКОНОМІЧНОЇ КІБЕРНЕТИКИ ТА
ІНФОРМАТИКИ**

Назва курсу	«Прикладне програмне забезпечення»
Викладач (-і)	Мушак Андрій Ярославович
Профайл викладача (-ів)	http://www.tneu.edu.ua/faculty/fkit/department-eki-fkit/staff-eki-fkit/6972-mushak-andrii-yaroslavovych.html
Контактний тел.	+380661717289
E-mail:	a.mushak@tneu.edu.ua
Сторінка курсу в moodle	https://moodle.tneu.edu.ua
Консультації	<i>Очні консультації:</i> п'ятниця: 11-00, ауд. 2210. <i>Он-лайн консультації:</i> у групі viber кожного дня з 11-00.

1. Коротка анотація до курсу

Теорія алгоритмів є теоретичним фундаментом програмування і всієї прикладної математики. Теорія алгоритмів безпосередньо пов'язана з теорією керування, вона є теоретичним фундаментом програмування й інформатики. Мови програмування базуються на уточненнях поняття алгоритму. Апарат математичної логіки і теорії алгоритмів необхідний для адекватного моделювання різноманітних предметних областей, створення сучасних програмних та інформаційних систем.

2. Пререквізити.

Необхідний мінімум для початку вивчення дисципліни – базові знання з дискретної математики, теорії множин, поняття про логіку та закони мислення, булеву алгебру.

Постреквізити. Дисципліна закладає знання та вміння, необхідні у процесі вивчення таких навчальних дисциплін, як бази даних та розподілені інформаційно-аналітичні системи, програмування та підтримка Веб-застосунків та ін.

3. Мета та цілі курсу

Базові поняття теорії алгоритмів, алгоритми і їх властивості, формальні аксіоматичні логічні системи, формальні моделі алгоритмів (нормальні алгоритми Маркова, машина Тюрінга, рекурсивні функції), алгоритмічно обчислювані функції, питання обчислюваності, проблеми розв'язності та нерозв'язності масових проблем.

Результати навчання:

Знання та вміння, набуті студентами при вивченні дисципліни «Теорія алгоритмів», необхідні їм також при вивченні дисциплін циклу професійної та практичної підготовки, зокрема, дисципліни «Моделювання систем», для аналізу матеріалів практик, виконання випускних кваліфікаційних робіт, в подальшій професійній діяльності тощо.

4. Загальна інформація про дисципліну

Ступінь вищої освіти	Бакалавр
Спеціальність	01539 – “Професійна освіта. Цифрові технології”
Курс (рік навчання)	1
Семестр	1
Рік викладання	2023
Формат курсу	Очний (<i>offline</i>)
Нормативна \ вибіркова	нормативна
Загальна кількість год/ кредитів	120/4
Лекції, год.	30
Семінарські заняття / практичні / лабораторні, год	30
Самостійна робота, год.	56

5. Перелік тем

1. Алгоритми. Формальні моделі алгоритмів
2. Системи Поста. Формальні граматики
3. Частково рекурсивні функції. Квазіарні ЧРФ, n-арні ЧРФ. Примітивні програмні алгебри. Примітивна мова програмування SIPL
4. Кодування, нумерації. Канторові нумерації. Універсальні класи алгоритмів. Теза Чорча
5. Нумерації ЧРФ. Теорема про параметризацію (s-m-n-теорема)
6. Рекурсивні та рекурсивно перелічні множини. Рекурсивні та частково рекурсивні предикати
7. Нерозв'язність, часткова розв'язність. Теореми Райса, Райса-Шапіро
8. Звідності. m-звідність. Продуктивні та креативні множини
9. Відносна обчислюваність. T-звідність
10. Арифметичність; теорема Тарського. Арифметична ієрархія
11. Ефективні оператори на функціях та множинах. Теореми Кліні про нерухому точку
12. Складність обчислень

6. Рекомендовані джерела інформації

1. Костюченко А. О. Основи програмування мовою Python : навч. посіб. Чернігів : ФОП Баликіна С. М., 2020. 180 с.
2. Васильєв О. М. Програмування мовою Python. Тернопіль : Навчальна книга – Богдан, 2019. 504 с.
3. Яковенко А. В. Основи програмування. Python. Частина 1 : підручник. Київ : КПІ ім. Ігоря Сікорського, 2018. 195 с.
4. Крєневич А. П. Python у прикладах і задачах. Частина 1. Структурне програмування : навч. посіб. Київ : ВПЦ "Київський Університет", 2017. 206 с.
5. Васильєв О. М. Програмування мовою Python. Тернопіль : Навчальна книга – Богдан, 2019. 504 с.

6. Reitz K., Schlusser T.: The Hitchhiker's Guide to Python: Best Practices for Development. O'Reilly Media, Inc., 2016.
7. Stephenson B.: The Python Workbook: A Brief Introduction with Exercises and Solutions, 2 nd ed. Springer, 2019.
8. Downey A.B.: Think Python: How to Think Like a Computer Scientist, 2nd ed. O'Reilly Media, Inc., 2016.
9. Phillips D.: Python 3 Object-oriented Programming: Unleash the power of Python 3 objects, 2 nd ed. Packt Publishing, 2015.
10. Kalb I.: Learn to Program with Python 3: A Step-by-Step Guide to Programming, 2nd ed. Apress, 2018.
11. Browning J.B., Alchin M.: Pro Python 3: Features and Tools for Professional Development, 3 rd ed. Apress, 2019.

7. Система оцінювання та вимоги

Підсумковий бал (за 100-бальною шкалою) з дисципліни “**Прикладне програмне забезпечення**” визначається як середньозважена величина, залежно від питомої ваги кожної складової залікового кредиту:

Заліковий модуль 1 - 20%

Заліковий модуль 2 (підсумкова контрольна робота) – 20%

Заліковий модуль 3 (оцінка за КППЗ, враховуючи поточне опитування) - 20%

Заліковий модуль 4 (екзамен) – 40%

Будь-яке завдання, за яке студент отримав оцінку, яка його не задовільняє може бути повторно перездано протягом наступних двох тижнів.

Шкала оцінювання:

За шкалою THEU	За національною шкалою	За шкалою ECTS
90–100	відмінно	A (відмінно)
85–89	добре	B (дуже добре)
75-84		C (добре)
65-74	задовільно	D (задовільно)
60-64		E (достатньо)
35-59	незадовільно	FX (незадовільно з можливістю повторного складання)
1-34		F (незадовільно з обов'язковим повторним курсом)

8. Навчальні ресурси

№	Найменування	Номер теми
1.	Середовища програмування Turbo Pascal, Embarcadero Delphi, пакет для математиків MathCad	1-12
2.	Мультимедійний проектор	1-12
3.	Проекційний екран	1-12
4.	Комунікаційне програмне забезпечення - браузерери	1-12

9. Політики курсу

Академічна доброчесність. Дотримання академічної доброчесності студентами передбачає:

- самостійне виконання навчальних завдань, завдань поточного та підсумкового контролю результатів навчання (для осіб з особливими освітніми потребами ця вимога застосовується з урахуванням їхніх індивідуальних потреб і можливостей);

- посилення на джерела інформації у разі використання ідей, розробок, тверджень, відомостей;

- дотримання норм законодавства про авторське право і суміжні права;

- надання достовірної інформації про результати власної навчальної (наукової, творчої) діяльності, використанні методики досліджень і джерела інформації.

Порушенням академічної доброчесності вважається:

академічний плагіат - оприлюднення (частково або повністю) наукових (творчих) результатів, отриманих іншими особами, як результатів власного дослідження (творчості) та/або відтворення опублікованих текстів (оприлюднених творів мистецтва) інших авторів без зазначення авторства;

самоплагіат - оприлюднення (частково або повністю) власних раніше опублікованих наукових результатів як нових наукових результатів;

фабрикація - вигадкування даних чи фактів, що використовуються в освітньому процесі або наукових дослідженнях;

фальсифікація - свідомо зміна чи модифікація вже наявних даних, що стосуються освітнього процесу чи наукових досліджень;

списування - виконання письмових робіт із залученням зовнішніх джерел інформації, крім дозволених для використання, зокрема під час оцінювання результатів навчання.

За порушення академічної доброчесності здобувачі освіти можуть бути притягнені до такої академічної відповідальності:

- повторне проходження оцінювання (контрольна робота, іспит, залік тощо);

- повторне проходження відповідного освітнього компонента освітньої програми.

Політика запізнення. За несвоєчасно виконані завдання буде накладено штраф 10 відсотків від загальної кількості балів за це завдання. Примітка. Виключення можуть бути зроблені до невчасно зданих завдань з поважних причин.