Код та назва дисципліни українською мовою/ Назва дисципліни англійською мовою	1-ф09-1 Методи організації та обробки даних у комп'ютерних системах / Methods of Data Organization and Processing in Computer Systems
Рекомендується для галузі знань *(спеціальності, освітньої програми)*	Для спеціальностей усіх галузей знань
Кафедра (*зазначати повну назву кафедри*)	Електронних обчислювальних машин
П.І.П. НПП *(за можливості)*	доцент, к.ф.-м.н. Карпенко Надія Валеріївна
Рівень ВО	Перший (бакалаврський)
Курс, семестр *(в якому буде викладатись)*	Курс: 2 (II семестр) 3-4 курси, будь-який семестр
Мова викладання	Українська
Пререквізити (передумови вивчення дисципліни)	Базові знання з програмування
Чому це цікаво/треба вивчати	• Знання структур даних (масиви, списки, дерева, графи) та алгоритмів сортування й пошуку дозволяє не лише вирішувати завдання, але й робити це оптимально, із мінімальними витратами ресурсів. • Добре знання структур даних і алгоритмів – це основа технічних інтерв'ю у IT-компаніях.
Перелік тем з дисциплін	✓ Вступ до алгоритмів і структур даних. Основні поняття та класифікація. Системи контролю версій. ✓ Аналіз ефективності алгоритмів. Оцінка часової та просторової складності. ✓ Лінійні структури даних: масиви, списки, операції та їх ефективність. ✓ Стек і черга: реалізація, застосування та оптимізація. ✓ Хеш-таблиці: принципи роботи, методи розв'язання колізій, оптимізація пошуку. ✓ Бінарні дерева пошуку: структура, операції вставки, видалення та пошуку. ✓ Балансовані дерева (AVL, червоно-чорні дерева): принципи балансування та оптимізації. ✓ Графи: представлення, основні алгоритми обходу (DFS, BFS), оптимізація роботи з графами. ✓ Алгоритми сортування. ✓ Пошук. ✓ Динамічне програмування. ✓ Рекурсія та рекурентні співвідношення: аналіз і оптимізація. ✓ Алгоритми роботи з рядками. ✓ Підходи до розробки алгоритмів: жадібні алгоритми, метод розділяй і володарюй.

	✓ Оптимізація коду: профілювання, виявлення "вузьких місць", покращення продуктивності. ✓ Паралельні алгоритми та оптимізація роботи з багатопоточністю. ✓ Методи зменшення використання пам'яті: стиснення даних, ефективні структури зберігання.
Як можна користуватися набутими знаннями і уміннями *(компетентності)*	• Отриманні знання можна використовувати у повсякденній розробці програмного забезпечення для створення якісних, конкурентоспроможних й ефективних цифрових продуктів. • Оптимізація програмного забезпечення стає важливою зі зростанням обсягів даних і складності систем. • Використання інструментів профілювання для виявлення "вузьких місць" у програмних модулях.
Очікувані результати навчання	• Дисципліна розвиває аналітичні навички, уміння розкладати складні завдання на прості підзадачі, будувати логічні залежності та шукати найкращі варіанти рішень. • Вміння оцінювати компроміси між продуктивністю, пам'яттю та складністю реалізації. • Оптимізація існуючих алгоритмів шляхом зменшення часової та просторової складності. • Опанування програмних засобів командної розробки ПЗ: Git, GitHub, Jira.
Інформаційне забезпечення	Методичний посібник для лабораторних робіт Комплект презентацій
Види навчальних занять *(лекції, практичні, семінарські, лабораторні заняття тощо)*	Лекції (28 год), лабораторні заняття (28 год)
Вид семестрового контролю	диф. залік
Максимальна кількість здобувачів на семестр/ Мінімальна кількість здобувачів *(тільки для мовних, творчих дисциплін, за необхідності)*	Без обмежень

Декан факультету _____ *Ігор ГОМІЛКО*