

Міністерство освіти і науки України
Національний аерокосмічний університет
«Харківський авіаційний інститут»

Кафедра «Інженерії програмного забезпечення» (№ 603)

ЗАТВЕРДЖУЮ

Гарант освітньої програми



(підпис)

Ілона ШЕВЧЕНКО

(ім'я та прізвище)

« 29 » 08 2025 р.

**СИЛАБУС ОBOB'ЯЗKОВОЇ
HAВЧАЛЬНОЇ ДИСЦИПЛІНИ**

Алгоритми і структури даних

(назва навчальної дисципліни)

Галузь знань: 12 Інформаційні технології

(шифр і найменування галузі знань)

Спеціальність: 121 Інженерія програмного забезпечення

(код та найменування спеціальності)

Освітня програма: Інженерія програмного забезпечення

(найменування освітньої програми)

Рівень вищої освіти: перший (бакалаврський)

Силабус введено в дію з 01.09.2025 року

Харків – 2025 р.

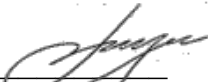
Розробник: Соколова Є.В., доц. каф. 603, канд. техн. наук., доц.
(прізвище та ініціали, посада, наукова ступінь та вчене звання)


(підпис)

Силабус навчальної дисципліни розглянуто на засіданні кафедри інженерії програмного забезпечення
(назва кафедри)

Протокол № 1 від « 29 » серпня 2025 р.

Завідувач кафедри д-р техн. наук., проф.
(науковий ступінь і вчене звання)


(підпис)

І.Б. Туркін
(ініціали та прізвище)

Погоджено з представником здобувачів освіти:
Представник студентського самоврядування


(підпис)

Дикун Д.В.
(ініціали та прізвище)

1. Загальна інформація про викладача



ПІБ: Соколова Євгенія Віталіївна

Посада: доцент кафедри інженерії програмного забезпечення

Науковий ступінь: кандидат технічних наук

Вчене звання: доцент

Перелік дисциплін, які викладає:

- Основи програмування
- Програмування мовою C#
- Алгоритми і структури даних

Напрями наукових досліджень:
інженерія програмного забезпечення, екосистеми програмного забезпечення та цифрові платформи

Контактна інформація:

e-mail: y.sokolova@khai.edu,

telegram: [@YevheniiaSokolova](https://www.instagram.com/YevheniiaSokolova)

2. Опис навчальної дисципліни

Форма здобуття освіти	<i>Денна</i>
Семестр	3 семестр
Мова викладання	Українська
Тип дисципліни	<i>Обов'язкова</i>
Обсяг дисципліни: кредити ЄКТС/ кількість годин	<u>денна</u> : 5 кредитів ЄКТС / 150 годин (64 аудиторних, з яких: лекції – 32, практичні – 32; СРЗ – 86)
Види навчальної діяльності	Лекції, практичні заняття, самостійна робота
Види контролю	Поточний контроль, модульний контроль, семестровий контроль – іспит
Пререквізити	«Дискретні структури», «Програмування мовою С#».

3. Мета та завдання навчальної дисципліни

Мета - надання студентам ґрунтовних знань та практичних навичок у галузі алгоритмів і структур даних, що дозволить їм ефективно розробляти, аналізувати та оптимізувати програмне забезпечення. Студенти навчатимуться обирати відповідні структури даних для розв'язання різноманітних задач, а також розробляти алгоритми, які забезпечують високу продуктивність і ефективність використання ресурсів

Завдання - вивчення впливу структур даних на ефективність виконання алгоритму, а також методів оцінки алгоритмів.

Компетентності, які набуваються:

Інтегральна компетентність:

Здатність розв'язувати складні спеціалізовані завдання або практичні проблеми інженерії програмного забезпечення, що характеризуються комплексністю та невизначеністю умов, із застосуванням теорій та методів інформаційних технологій.

Загальні компетентності (ЗК)

Після закінчення цієї програми здобувач освіти буде здатен:

ЗК01. Здатність до абстрактного мислення, аналізу та синтезу.

ЗК02. Здатність застосовувати знання у практичних ситуаціях.

ЗК05. Здатність вчитися і оволодівати сучасними знаннями.

ЗК06. Здатність до пошуку, оброблення та аналізу інформації з різних джерел.

ЗК13. Здатність ухвалювати рішення та діяти, дотримуючись принципу неприпустимості корупції та будь-яких інших проявів недоброчесності.

Фахові компетентності

Після закінчення цієї програми здобувач освіти буде здатен:

ФК07. Володіння знаннями про інформаційні моделі даних, здатність створювати програмне забезпечення для зберігання, видобування та опрацювання даних.

ФК13. Здатність обґрунтовано обирати та освоювати інструментарій з розробки та супроводження програмного забезпечення.

ФК14. Здатність до алгоритмічного та логічного мислення.

ФК16. Здатність розробляти методичні, інформаційні, математичні, алгоритмічні та програмні засоби реалізації інформаційних технологій

Програмні результати навчання:

ПРН01. Аналізувати, цілеспрямовано шукати і вибирати необхідні для вирішення професійних завдань інформаційно-довідникові ресурси і знання з урахуванням сучасних досягнень науки і техніки.

ПРН05. Знати і застосовувати відповідні математичні поняття, методи доменного, системного і об'єктно-орієнтованого аналізу та математичного моделювання для розробки програмного забезпечення.

ПРН07. Знати і застосовувати на практиці фундаментальні концепції, парадигми і основні принципи функціонування мовних, інструментальних і обчислювальних засобів інженерії програмного забезпечення.

ПРН13. Знати і застосовувати методи розробки алгоритмів, конструювання програмного забезпечення та структур даних і знань.

ПРН15. Мотивовано обирати мови програмування та технології розробки для розв'язання завдань створення і супроводження програмного забезпечення.

ПРН27. Вміти використовувати відомі алгоритми та чисельні методи для розробки програмних застосунків.

4. Зміст навчальної дисципліни

Змістовий модуль 1.

Базові поняття теорії алгоритмів і структури даних

Тема 1. Вступ. Місце та значення курсу. Базові поняття теорії алгоритмів

Анотація: Вступ. Місце та значення курсу. Загальні поняття. Місце та значення курсу для інженерів-програмістів. Базові поняття теорії алгоритмів: визначення алгоритму; властивості алгоритмів; обчислювальна складність; класи алгоритмів. Поняття структури даних: рівні опису структур даних: абстрактний (функціональна специфікація), логічний рівень та фізичний; моделі даних; моделі пам'яті; типи даних у мовах програмування; структура даних як об'єднання типу даних і моделі пам'яті.

Тема лекції 1: Вступ. Місце та значення курсу. Загальні поняття.

Самостійна робота здобувачів: опрацювання матеріалу лекцій, формування питань до викладача.

Тема 2. Структурні та лінійні типи даних

Анотація: Структурні та лінійні типи даних: масив, множина, запис, таблиця, списки, стек, черга, дек. Реалізація списків з послідовним розміщенням. Реалізація списків з динамічним розміщенням. Порівняння послідовного і динамічного розміщення. Поняття АДЗ зв'язний список. Класифікація АДЗ зв'язний список: однозв'язний, двозв'язний, циклічний, багатозв'язний. Опис АДЗ зв'язний список. Фізичне представлення зв'язного списку. черги. Способи усунення недоліків. Приклади реалізації стека і черги.

Тема лекції 1: Структурні та лінійні типи даних.

Самостійна робота здобувачів: опрацювання матеріалу лекцій, формування питань до викладача.

Тема 3. Реалізації стеку та черги

Анотація: Реалізації стеку та черги з послідовним розміщенням та динамічним розміщенням мовою C#. Практичне застосування колекцій.

Тема лекції 1: Реалізації стеку та черги.

Тема практичного заняття 1: Розробка абстрактного типу даних та діаграми класів до індивідуальної практичної роботи 1 «Базові динамічні структури даних. Лінійні списки».

Тема практичного заняття 2: Виконання індивідуальної практичної роботи 1 «Базові динамічні структури даних. Лінійні списки»

Самостійна робота здобувачів: опрацювання матеріалу лекцій, підготовка до практичних занять, проходження тестування за результатами роботи на практичних заняттях, формування звіту з індивідуальної практичної роботи 1, формування питань до викладача.

Тема 4. Вбудовані колекції

Анотація: Вбудовані колекції. Базові інтерфейси IEnumerable та IQueryable. Ієрархія вбудованих класів. Вбудовані класи Stack, Stack<T>, Queue, Queue<T>, ArrayList, List<T>, LinkedList<T>, основні операції з цими класами. Приклади обробки.

Тема лекції 1: Вбудовані колекції.

Тема лекції 2: Вбудовані класи та основні операції з цими класами.

Самостійна робота здобувачів: опрацювання матеріалу лекцій, формування питань до викладача.

Тема 5. Циклічні списки

Анотація: Циклічні списки. Операції на циклічних списках. Обробка елементів циклічного списку. Приклади реалізації стека і черги на циклічних списках.

Тема лекції 1: Циклічні списки.

Тема практичного заняття 1: Розробка абстрактного типу даних та діаграми класів до індивідуальної практичної роботи 2 «Циклічні списки».

Тема практичного заняття 2: Виконання індивідуальної практичної роботи 2 «Циклічні списки»

Самостійна робота здобувачів: опрацювання матеріалу лекцій, підготовка до практичних занять, проходження тестування за результатами роботи на практичних заняттях, формування звіту з індивідуальної практичної роботи 2, формування питань до викладача.

Тема 6. Дворівневі структури

Анотація: Дворівневі структури. Реалізація списку підписків на базі вбудованих колекцій C#.

Тема лекції 1: Дворівневі структури

Тема практичного заняття 1: Розробка абстрактного типу даних до індивідуальної практичної роботи 3 «Дворівневі структури. Списки списків. Вбудовані колекції».

Тема практичного заняття 2: Розробка діаграми класів до індивідуальної практичної роботи 3 «Дворівневі структури. Списки списків. Вбудовані колекції».

Тема практичного заняття 3: Виконання індивідуальної практичної роботи 3 «Дворівневі структури. Списки списків. Вбудовані колекції».

Самостійна робота здобувачів: опрацювання матеріалу лекцій, підготовка до практичних занять, проходження тестування за результатами роботи на практичних заняттях, формування звіту з індивідуальної практичної роботи 3, формування питань до викладача.

Тема 7. Хеш-таблиці

Анотація: Hash-таблиці. Поняття хеш-таблиця, властивості хеш-таблиць. Проблема колізій у хеш-таблицях та способи їх вирішення. Хеш-функції. Клас Hashtable і Dictionary, основні властивості і методи. Поняття еквівалентності об'єктів. Методи Equals та GetHashCode, імутабельність класів. Приклади роботи з Hashtable та Dictionary. Приклади реалізації.

Тема лекції 1: Hash-таблиці.

Тема лекції 2: Клас Hashtable і Dictionary

Самостійна робота здобувачів: опрацювання матеріалу лекцій, формування питань до викладача.

Модульний контроль 1

Форма занять: написання модульної роботи в аудиторії (за рішенням лектора допускається проведення у дистанційній формі).

Змістовий модуль 2.

*Дерева. Графи. Алгоритми сортування і пошуку.
Оцінка ефективності та трудомісткості алгоритмів*

Тема 8. Нелінійні структури

Анотація: Дерева. Бінарні дерева. Фізичне представлення дерев в послідовній та динамічній пам'яті. Алгоритм перетворення m-арного дерева в бінарне. Основні операції з бінарними деревами. Обходи вузлів дерева в прямому, зворотному і кінцевому порядку. Приклади використання дерев. Приклади реалізації дерев.

Тема лекції 1: Дерева. Обходи дерев

Тема лекції 2: Приклади реалізації дерев.

Тема практичного заняття 1: Розробка m-арного дерева до індивідуальної практичної роботи 4 «Дерева. Форми подання дерев. Обходи дерев». Перетворення в бінарне дерево. Показ представлення дерева у пам'яті.

Тема практичного заняття 2: Розробка діаграми класів та виконання індивідуальної практичної роботи 4 «Дерева. Форми подання дерев. Обходи дерев».

Самостійна робота здобувачів: опрацювання матеріалу лекцій, підготовка до практичних занять, проходження тестування за результатами роботи на практичних заняттях, формування звіту з індивідуальної практичної роботи 4, формування питань до викладача.

Тема 9. Специфічні види дерев

Анотація: Специфічні види дерев. Збалансовані дерева. Алгоритм створення збалансованого дерева. Використання дерев для пошуку інформації. Бінарні дерева пошуку (BST). AVL-дерева. Основні операції з AVL-деревами. Червоно-чорні дерева (RB). Властивості і основні операції з RB-деревами. Приклади реалізації. Прошиті дерева. Типи прошитих дерев. Косе дерево. Приклади реалізації. B-дерева. B+-дерева. Графи. Форми подання графів в ЕОМ. Купа.

Тема лекції 1: Специфічні види дерев.

Тема лекції 2: Графи.

Тема практичного заняття 1: Розробка графа до індивідуальної практичної роботи 5 «Графи. Форми подання графів в ЕОМ». Показ представлення графа у пам'яті.

Тема практичного заняття 2: Розробка діаграми класів та виконання індивідуальної практичної роботи 5 «Графи. Форми подання графів в ЕОМ».

Самостійна робота здобувачів: опрацювання матеріалу лекцій, підготовка до практичних занять, проходження тестування за результатами роботи на практичних заняттях, формування звіту з індивідуальної практичної роботи 5, формування питань до викладача.

Тема 10. Сортування

Анотація: Сортування. Визначення задачі сортування. Сортування вибіркою. Сортування включенням. Сортування розподілом. Сортування злиттям. Швидке сортування. Порівняння методів сортування за витратами часу та пам'яті.

Тема лекції 1: Внутрішні алгоритми сортування.

Тема лекції 2: Зовнішні алгоритми сортування.

Тема практичного заняття 1: Розробка алгоритму сортування до індивідуальної практичної роботи 6 «Сортування простих структур даних».

Тема практичного заняття 2: Розробка діаграми класів та виконання індивідуальної практичної роботи 6 «Сортування простих структур даних».

Самостійна робота здобувачів: опрацювання матеріалу лекцій, підготовка до практичних занять, проходження тестування за результатами роботи на практичних заняттях, формування звіту з індивідуальної практичної роботи 6, формування питань до викладача.

Тема 11. Пошук

Анотація: Пошук. Загальна класифікація алгоритмів пошуку. Лінійний пошук. Бінарний пошук. Пошук методом Фібоначчі. Методи пошуку з обчисленням адреси (хеш-таблиці). Алгоритми пошуку в бінарному дереві (BST, DFS). Алгоритми пошуку підпоследовності. Порівняння методів пошуку за витратами часу та пам'яті.

Тема лекції 1: Алгоритми пошуку.

Тема практичного заняття 1: Розробка абстрактного типу даних до індивідуальної практичної роботи 7 «Робота з Dictionary. Пошук».

Тема практичного заняття 2: Розробка діаграми класів до індивідуальної практичної роботи 7 «Робота з Dictionary. Пошук».

Тема практичного заняття 3: Виконання індивідуальної практичної роботи 7 «Робота з Dictionary. Пошук».

Самостійна робота здобувачів: опрацювання матеріалу лекцій, підготовка до практичних занять, проходження тестування за результатами роботи на практичних заняттях, формування звіту з індивідуальної практичної роботи 6, формування питань до викладача.

Модульний контроль 2

Форма занять: написання модульної роботи в аудиторії (за рішенням лектора допускається проведення у дистанційній формі).

5. Індивідуальні завдання

Виконання розрахункової роботи на тему «ПОРІВНЯЛЬНИЙ АНАЛІЗ ПРОДУКТИВНОСТІ РОБОТИ СТРУКТУР ДАНИХ ТА ДОСЛІДЖЕННЯ АЛГОРИТМУ СОРТУВАННЯ»

6. Методи навчання

Лекції з елементами інтерактиву (пояснення з використанням презентацій, прикладів коду, міні-опитувань). *Практичні заняття* – розробка програм у середовищах програмування, розв’язування задач у командах та індивідуально. *Проектно-орієнтоване навчання* – виконання невеликих практичних проєктів, спрямованих на закріплення знань. *Робота в малих групах* – колективний аналіз програмних фрагментів, обговорення рішень. *Використання системи онлайн-тестування.* *Самостійна робота* – індивідуальні завдання, робота з електронними матеріалами та онлайн-курсами. *Консультації* – індивідуальні та групові (очно або онлайн) для підтримки та корекції навчального процесу.

7. Методи контролю

Поточний контроль: опитування на практичних заняттях; розв’язування алгоритмічних задач та аналіз програмних фрагментів; виконання письмових контрольних робіт з окремих розділів курсу; програмований контроль (тестування, онлайн-тести); оцінювання виконання індивідуальних і групових практичних завдань.

Модульний контроль: складання модульного контролю;

Підсумковий контроль: іспит.

8. Критерії оцінювання та розподіл балів, які отримують здобувачі

Складові навчальної роботи	Бали за одне заняття (завдання)	Кількість занять (завдань)	Сумарна кількість балів
Змістовний модуль 1			
Робота на лекціях			
Робота на практичних заняттях			
Виконання і захист практичних робіт	6...9	3	18...27
Модульний контроль	6...14	1	6...14
Змістовний модуль 2			
Робота на лекціях			
Робота на практичних заняттях			
Виконання і захист практичних робіт	6...9	4	24...36
Модульний контроль	6...1	1	6...14
Виконання і захист РГР (РР, РК)	6..9	1	6..9
Усього за семестр			60..100

Допуском до семестрового контролю є отримання позитивної оцінки з усіх практичних робіт і розрахункової.

Семестровий контроль (іспит) проводиться у разі відмови студента від балів поточного тестування й за наявності допуску до іспиту. Під час складання семестрового іспиту студент має можливість отримати максимум 100 балів.

Білет для іспиту складається з трьох завдань:

1. Підсумкове тестування (50 балів).
2. Розв'язання 2 задач. Для кожної задачі треба визначити вхідні/вихідні данні, скласти алгоритм у вигляді блок-схеми та написати програму (50 балів).

Таблиця 8.2 – Шкали оцінювання: бальна і традиційна

Сума балів	Оцінка за традиційною шкалою	
	Іспит, диференційний залік	Залік
90 – 100	Відмінно	Зараховано
75 – 89	Добре	
60 – 74	Задовільно	
0 – 59	Незадовільно	Не зараховано

Критерії оцінювання роботи здобувача протягом семестру

Задовільно (60-74). Показати мінімум знань та умінь. Здати індивідуальне завдання та практичні роботи 1,2,6. Здати тестування. Знати методи побудови алгоритмів; фундаментальні алгоритми (сортування, пошуку, на графах); базові структури даних. Вміти вибирати для конкретної задачі структури даних, що забезпечують можливість побудови алгоритмів.

Добре (75-89). Твердо знати мінімум, здати практичні роботи 1, 2, 3, 4, 6 індивідуальне завдання та тестування. Досконало знати методи побудови алгоритмів; фундаментальні алгоритми (сортування, пошуку, на графах); базові структури даних, абстрактні структури даних. Досконало вміти вибирати для конкретної задачі структури даних та мовні конструкції, що забезпечують можливість побудови ефективних алгоритмів.

Відмінно (90-100). Здати всі контрольні точки з оцінкою «відмінно». Досконально знати всі теми та вміти застосовувати їх.

9. Політика навчального курсу

Процедура відпрацювання пропущених занять: здобувач самостійно ознайомлюється з пропущеним матеріалом: лекційними конспектами, навчальними презентаціями, записами занять або додатковими матеріалами, наданими викладачем. Пропущене практичне заняття відпрацьовується шляхом виконання всіх завдань, передбачених для цього, у середовищі Microsoft Visual Studio 2022 (MVS2022). За потреби здобувач може узгодити індивідуальну консультацію з викладачем для роз'яснення складних тем або перевірки виконаних завдань. Після виконання завдань здобувач надає результати викладачу для перевірки у форматі програмного коду та звіту про виконану роботу. Виконане заняття оцінюється за тими ж критеріями, що і основне заняття. Відпрацювання вважається успішним після схвалення викладачем результатів та підтвердження засвоєння матеріалу.

Дотримання вимог академічної доброчесності здобувачами освіти під час вивчення навчальної дисципліни. Під час вивчення навчальної дисципліни здобувачі освіти мають дотримуватися загальноприйнятих морально-етичних норм і правил поведінки, вимог академічної доброчесності, передбачених Положенням про академічну доброчесність Національного аерокосмічного університету «Харківський авіаційний інститут» (<https://khai.edu/assets/files/polozhennya/polozhennya-pro-akademichnu->

dobrochesnist.pdf). Очікується, що розроблені здобувачами програми та звіти до них будуть оригінальними. Відсутність посилань на використані джерела, фабрики джерел, списування, втручання в роботу інших здобувачів освіти становлять, але не обмежують, приклади можливої академічної недоброчесності. Виявлення ознак академічної недоброчесності в письмовій роботі здобувача освіти є підставою для її незарахування викладачем незалежно від масштабів плагиату чи обману.

Вирішення конфліктів. Порядок і процедури врегулювання конфліктів, пов'язаних із корупційними діями, зіткненням інтересів, різними формами дискримінації, сексуальними домаганнями, міжособистісними стосунками та іншими ситуаціями, що можуть виникнути під час навчання, а також правила етичної поведінки регламентуються Кодексом етичної поведінки в Національному аерокосмічному університеті «Харківський авіаційний інститут» (<https://khai.edu.ua/university/normativna-baza/ustanovchi-dokumenti/kodeks-etichnoi-povedinki/>).

10. Методичне забезпечення та інформаційні ресурси

1. Дистанційний курс дисципліни розроблено у системі дистанційного навчання Mentor, яку впроваджено в Національному аерокосмічному університеті ім. М.Є. Жуковського «ХАІ», доступ до курсу за посиланням: <https://mentor.khai.edu/course/view.php?id=238>

Інформаційні ресурси:

2. Довідкові матеріали з Visual C#.
<https://docs.microsoft.com/dotnet/csharp/language-reference/>

11. Рекомендована література

Базова

1. Алгоритми та структури даних: навч. Посібник / О. Коротєєва. Львів: Видавництво Львівської політехніки, 2014. 280 с.
2. Крєневич А.П. Алгоритми і структури даних. Підручник. – К.: ВПЦ "Київський Університет", 2021. – 200 с.
3. Шаховська Н.Б., Голощук Р.О. Алгоритми і структури даних: навч. посібник / О. Коротєєва. Львів: Видавництво "Магнолія 2006" Львів, 2023. 216 с.
4. <https://learn.microsoft.com/en-us/dotnet/csharp/tour-of-csharp/tutorials/>

Допоміжна

1. Geoge Heieman. Learning Algorithms: A Programmer's Guide to Writing Better Code 1st Edition / O'Reilly Media, 2021. – 278 p.
2. Data Structure Visualizations
<https://www.cs.usfca.edu/~galles/visualization/Algorithms.html>

12. Інформаційні ресурси

1. Інтегроване середовище розробки: Visual Studio Community <https://visualstudio.microsoft.com/vs/community>
2. <https://dotnet.microsoft.com/en-us/learn/csharp>

Зміст силабуса адаптовано до поточних вимог роботодавців і валідовано за результатами навчання студентів у 2024/2025 році за участі ІТ-компаній Kharkiv IT Cluster у межах проєкту «Система сертифікації ІТ-дисциплін»

