

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

Кафедра математичного моделювання та штучного інтелекту (№ 304)

ЗАТВЕРДЖУЮ

Гарант освітньої програми


(підпис)

Дмитро ЧУМАЧЕНКО
(ім'я та ПРІЗВИЩЕ)

« 21 » 08 2025 р.

**СИЛАБУС ОБОВ'ЯЗКОВОЇ
НАВЧАЛЬНОЇ ДИСЦИПЛІНИ**

Об'єктно-орієнтоване програмування (мова C#)
(назва навчальної дисципліни)

Галузь знань: 12 «Інформаційні технології»
(шифр і найменування галузі знань)

Спеціальність: 122 «Комп'ютерні науки»
(код і найменування спеціальності)

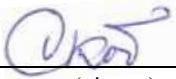
Освітня програма: «Інтелектуальні системи та технології»
(найменування освітньої програми)

Рівень вищої освіти: *перший (бакалаврський)*

Силабус уведено в дію з 01.09.2025

Харків – 2025 р.

Розробник: Скоб Ю.О., проф. каф. 304, д. т. н., проф.
(прізвище та ініціали, посада, науковий ступінь і вчене звання)


(підпис)

Силабус навчальної дисципліни розглянуто на засіданні кафедри № 304
математичного моделювання та штучного інтелекту
(назва кафедри)

Протокол № 1 від « 21 » серпня 2025 р.

В. о. завідувача кафедри к.ф.-м.н., доцент
(науковий ступінь і вчене звання)


(підпис)

Олексій КАРТАШОВ
(ім'я та ПРИЗВИЩЕ)

Погоджено з представником здобувачів освіти:

Здобувач першого (бакалаврського)

рівня вищої освіти за ОП

«Інтелектуальні системи та технології»


(підпис)

Катерина ПИЛИПЕНКО
(ім'я та ПРИЗВИЩЕ)

1. Загальна інформація про викладача



ПІБ: *Скоб Юрій Олексійович*

Посада: *професор каф. 304*

Науковий ступінь: *докт. техн. наук*

Вчене звання: *професор*

Перелік дисциплін, які викладає:

Методи обчислень; Об'єктно-орієнтоване програмування (мова C#); Паралельні та розподілені обчислювання; Розподілені інформаційно-аналітичні системи

Напрями наукових досліджень:

Математичне моделювання наслідків впливу техногенних аварій, пов'язаних з викидом небезпечних газів у атмосферу, на навколишнє середовище

Контактна інформація:

y.skob@khai.edu

2. Опис навчальної дисципліни

Форма здобуття освіти	Денна
Семестр	3-й
Мова викладання	Українська
Тип дисципліни	Обов'язкова
Обсяг дисципліни: кредити ЄКТС/ кількість годин	4,0 кредита ЄКТС / 120 годин (64 аудиторних, з яких: лекції – 32, практичні – 32; СРЗ – 56);
Види навчальної діяльності	Лекції, практичні заняття, робота контрольна (РК), самостійна робота
Види контролю	Поточний контроль, модульний контроль, семестровий контроль – іспит
Пререквізити	«Програмування та алгоритмічні мови», «Основи програмування (мова C++)», «Організація та обробка електронної інформації»

3. Мета та завдання навчальної дисципліни, переліки компетентностей та очікуваних результатів навчання

Мета – формування у студентів системних знань і практичних навичок розроблення програмного забезпечення із застосуванням принципів об'єктно-орієнтованого підходу на прикладі мови програмування C# в інтегрованому середовищі Visual Studio на платформі .NET. Курс спрямований на засвоєння концепцій інкапсуляції, успадкування, поліморфізму, абстракції даних, а також сучасних засобів проєктування, тестування та налагодження об'єктно-орієнтованих застосунків.

Завдання – поетапне ознайомлення здобувачів освіти з принципами, концепціями та технологіями об'єктно-орієнтованого підходу до проєктування програмного забезпечення, формування розуміння архітектури платформи .NET та ролі мови C# у створенні сучасних застосунків, набуття навичок роботи з базовими та складеними типами даних, класами, об'єктами, властивостями, методами, подіями та винятками, розвиток умінь застосовувати принципи інкапсуляції, успадкування, поліморфізму та абстракції для побудови гнучких і розширюваних програмних систем, опанування технологій роботи з колекціями, файлами, потоками, графічним інтерфейсом користувача (GUI) та бібліотеками класів .NET, формування навичок використання середовища розробки Visual Studio для створення, тестування та налагодження об'єктно-орієнтованих програм, розвиток умінь командної розробки застосунків з використанням систем контролю версій (Git) і засобів спільної роботи над кодом, підготовка до подальшого вивчення дисциплін, пов'язаних із розробленням корпоративних, веб- і мобільних застосунків мовою C#.

Компетентності, які набуваються:

Інтегральна компетентність:

Здатність розв'язувати складні спеціалізовані задачі та практичні проблеми у галузі комп'ютерних наук або у процесі навчання, що передбачає застосування теорій та методів інформаційних технологій і характеризується комплексністю та невизначеністю умов.

Загальні компетентності (ЗК)

Після закінчення цієї програми здобувач освіти буде здатен:

- до абстрактного мислення, аналізу та синтезу (ЗК 1);
- спілкуватися державною мовою як усно, так і письмово (ЗК 4);
- вчитися й оволодівати сучасними знаннями (ЗК 6);
- генерувати нові ідеї (креативність) (ЗК 8);
- приймати обґрунтовані рішення (ЗК 11);
- оцінювати та забезпечувати якість виконуваних робіт (ЗК 12).

Спеціальні компетентності (ФК)

Після закінчення цієї програми здобувач освіти буде здатен:

- до логічного мислення, побудови логічних висновків, використання формальних мов і моделей алгоритмічних обчислень, проектування, розроблення й аналізу алгоритмів, оцінювання їх ефективності та складності, розв'язності та нерозв'язності алгоритмічних проблем для адекватного моделювання предметних областей і створення програмних та інформаційних систем (СК 3);
- проектувати та розробляти програмне забезпечення із застосуванням різних парадигм програмування: узагальненого, об'єктно-орієнтованого, функціонального, логічного, з відповідними моделями, методами й алгоритмами обчислень, структурами даних і механізмами управління (СК 8).

Програмні результати навчання (ПРН):

- Розробляти моделі предметних середовищ, вибирати парадигму програмування з позицій зручності та якості застосування для реалізації методів та алгоритмів розв'язання задач в галузі комп'ютерних наук (ПРН 9);
- Використовувати інструментальні засоби розробки клієнт-серверних застосувань, проектувати концептуальні, логічні та фізичні моделі баз даних, розробляти та оптимізувати запити до них, створювати розподілені бази даних, сховища та вітрини даних, бази знань, у тому числі на хмарних сервісах, із застосуванням мов веб-програмування (ПРН 10);
- Володіти навичками управління життєвим циклом програмного забезпечення, продуктів і сервісів інформаційних технологій відповідно до вимог і обмежень замовника, вміти розробляти проектну документацію (техніко-економічне обґрунтування, технічне завдання, бізнес-план, угоду, договір, контракт) (ПРН 11);
- Володіти мовами системного програмування та методами розробки програм, що взаємодіють з компонентами комп'ютерних систем, знати мережні технології, архітектури комп'ютерних мереж, мати практичні навички технології адміністрування комп'ютерних мереж та їх програмного забезпечення (ПРН 13);
- Застосовувати знання методології та CASE-засобів проектування складних систем, методів структурного аналізу систем, об'єктно-орієнтованої методології проектування при розробці і дослідженні функціональних моделей організаційно-економічних і виробничо-технічних систем (ПРН 14).

4. Зміст навчальної дисципліни

МОДУЛЬ 1

Змістовний модуль 1. Структурне програмування мовою C#

Тема 1. Предмет вивчення і задачі дисципліни

Стисла анотація: Предмет і об'єкт дослідження, цілі і задачі дисципліни «Об'єктно-орієнтоване програмування (мова C#)». Основні

історичні етапи розвитку інформатики та програмування. Ознайомлення з інтегрованим середовищем програмування Visual Studio. Основні принципи структурного програмування. Складання алгоритму рішення задачі шляхом її декомпозиції на основі спадної і/або висхідної концепцій. Приклади побудови блок-схем.

Теми лекцій і практичних занять: «Інтегроване середовище Visual Studio C#».

Самостійна робота здобувача освіти: опрацювання матеріалу лекції, інсталяція інтегрованого середовища Visual Studio C# з офіційного сайту Microsoft, виконання індивідуальних завдань по інсталяції середовища та перевірки його працездатності створювати простий консольний проект виведення інформації про себе на екран, підготовка до захисту лабораторної роботи.

Тема 2. Стандартні типи даних. Вирази

Стисла анотація: Алфавіт мови Visual C#. Структура програми мовою Visual C#. Коментарі. Вбудовані типи даних. Декларація та використання констант і змінних в програмі. Перетворення типів. Процедури найпростішого безформатного введення даних з клавіатури і виведення на екран. Формати введення/виведення даних. Визначення виразу в Visual C#. Арифметичні, логічні та символічні вирази. Основні операції в виразах і їх пріоритет. Оператор присвоєння. Ініціалізація даних в програмі. Директиви компілятора.

Теми лекцій і лабораторного заняття: «Вирази C#».

Самостійна робота здобувача освіти: опрацювання матеріалу лекції, виконання індивідуального завдання з програмування арифметичних, логічних і рядкових виразів мовою C# відповідно варіанту, підготовка до захисту лабораторної роботи.

Тема 3. Розгалуження алгоритму

Стисла анотація: Умовний оператор if (скорочена та повна форми). Мітки й оператори переходу goto. Розгалуження обчислювального процесу з використанням оператора switch.

Теми лекцій і практичних занять: «Умовні інструкції C#».

Самостійна робота здобувача освіти: опрацювання матеріалу лекцій, виконання індивідуальних завдань використання інструкцій розгалуження обчислювального процесу у консольних додатках мовою C# відповідно варіанту, підготовка до захисту лабораторної роботи.

Тема 4. Оператори циклічної обробки. Обробка масивів. Файли даних

Стисла анотація: Циклічні оператори (for, foreach, while, do while). Організація циклів з передумовою та умовою в кінці тіла циклу. Приклади циклічних алгоритмів. Оператори виходу із циклу (break, continue). Застосування різних видів циклу під час знаходження суми безкінечного ряду.

Визначення масивів. Способи декларації масивів (в операторах об'яви типу). Ініціалізація масиву. Розташування елементів у пам'яті ЕОМ. Обробка одновимірних і багатовимірних масивів. Типові алгоритми пошуку екстремумів в масивах. Впорядкування масивів. Файлові типи даних. Стандартні файли і пристрої. Стандартні процедури для роботи з файлами. Файли послідовної організації і доступу. Текстові файли. Файли прямого доступу.

Теми лекції і лабораторного заняття: «Інструкції циклу C#».

Самостійна робота здобувача освіти: опрацювання матеріалу лекцій, виконання індивідуального завдання циклічної обробки даних, організованих у вигляді масивів та файлів C#, відповідно варіанту, підготовка до захисту лабораторної роботи.

Тема 5. Стандартні функції. Функції користувача

Стисла анотація: Доцільність використання функцій. Декларація стандартних функцій в програмі. Принцип локалізації імен. Формальні та фактичні параметри, правила їх співвідношення. Механізм передачі параметрів. Оператори виклику функцій. Приклади використання стандартних функцій. Глобальні функції. Декларація і визначення функцій користувача в програмі. Спільні риси та відмінності стандартних функцій і функцій користувача. Правила виклику функцій. Повернення результату роботи функції. Механізм передачі параметрів. Приклади написання функцій користувача.

Теми лекції і практичних занять: «Функції C#».

Самостійна робота здобувача освіти: опрацювання матеріалу лекцій, виконання індивідуальних завдань з використанням відносно незалежних програмних одиниць у вигляді функцій (методів) відповідно варіанту, підготовка до захисту лабораторної роботи.

Модульний контроль 1

Змістовний модуль 2. Об'єктно-орієнтоване програмування

Тема 1. Класи C#

Стисла анотація: Доцільність використання класів. Структура класу. Принципи об'єктно-зорієнтованого програмування (інкапсуляція, успадкування і поліморфізм). Декларація і визначення класів. Інтерфейси. Визначення і декларація типу. Поняття об'єкту. Доступ до методів класу. Властивості і їх визначення та поведінка. Сектори доступу до елементів класу. Принцип інкапсуляції.

Теми лекцій і практичних занять: «Класи C#».

Самостійна робота здобувача освіти: опрацювання матеріалу лекцій, виконання індивідуальних завдань і організації обчислювального процесу із застосуванням принципу інкапсуляції властивостей і поведінки об'єкта у класі як типі користувача відповідно варіанту, підготовка до захисту лабораторної роботи.

Тема 2. Спадкування і поліморфізм C#

Стисла анотація: Спадкування методів та змінних класів. Побудова дерев успадкування. Принцип повторного використання коду. Доступ до членів батьківського класу. Визначення поліморфної поведінки об'єктів. Конкретний і класичний поліморфізм. Використання поліморфізму для написання компактного коду. Поліморфізм на основі базового класу. Використання абстрактних класів. Поліморфізм на основі інтерфейсів. Поліморфізм на основі виклику методів.

Теми лекцій і практичних занять: «Спадкування і поліморфізм C#».

Самостійна робота здобувача освіти: опрацювання матеріалу лекцій, виконання індивідуальних завдань (побудова дерев класів і повторне використання коду; використання можливостей поліморфної поведінки екземплярів споріднених класів) відповідно варіанту, підготовка до захисту лабораторної роботи.

Тема 3. Вкладені класи C#

Стисла анотація: Визначення внутрішніх (вкладених) класів. Доступ до членів внутрішніх класів. Принципи використання внутрішніх класів.

Теми лекцій і практичних занять: «Вкладені класи C#».

Самостійна робота здобувача освіти: опрацювання матеріалу лекцій, виконання індивідуальних завдань з використанням вкладених класів C# відповідно варіанту, підготовка до захисту лабораторної роботи.

Тема 4. Структури C#

Стисла анотація: Доцільність використання структур. Оголошення і визначення структури. Принцип об'єктно-зорієнтованого програмування інкапсуляції властивостей об'єкта через структуру. Поняття об'єкту структури. Доступ до полів структури.

Теми лекцій і практичних занять: «Структури C#».

Самостійна робота здобувача освіти: опрацювання матеріалу лекцій, виконання індивідуальних завдань і організації обчислювального процесу із застосуванням принципу інкапсуляції властивостей у полях структури як складному типі користувача відповідно варіанту, підготовка до захисту лабораторної роботи.

Тема 5. Перевантаження операцій C#

Стисла анотація: Синтаксис перевантаження операторів. Перевантаження унарних операторів (інкремент, декремент). Перевантаження бінарних операторів. Аргументи та результати, що повертаються.

Теми лекцій і практичних занять: «Перевантаження операцій C#».

Самостійна робота здобувача освіти: опрацювання матеріалу лекцій, виконання індивідуальних завдань з використанням перевантаження операцій відповідно варіанту, підготовка до захисту лабораторної роботи.

Тема 6. Масиви та колекції C#

Стисла анотація: Правила оголошення масивів, створення об'єктів класу System.Array. Використання методів і властивостей об'єктів масивів. Ініціалізація елементів масивів і використання одновимірних і багатовимірних масивів C#. Вивчення функціонування оператора foreach для обробки колекцій.

Теми лекцій і практичних занять: «Масиви C#».

Самостійна робота здобувача освіти: опрацювання матеріалу лекцій, виконання індивідуальних завдань з використанням масивів як колекції елементів одного типу відповідно варіанту, підготовка до захисту лабораторної роботи.

Тема 7. Виключення C#

Стисла анотація: Визначення виключень. Правила обробки виключень. Структура, призначення і правила використання блоків try / catch / finally. Об'єкти виключень. Виклики стандартних, призначених для користувача і неспеціфікованих виключень. делегатів, як засоби програмування подій на мові C #; освоєння правил оголошення делегатів і створення екземплярів делегатів.

Теми лекцій і практичних занять: «Виключення C#».

Самостійна робота здобувача освіти: опрацювання матеріалу лекцій, виконання індивідуальних завдань з використанням виключень відповідно варіанту, підготовка до захисту лабораторної роботи.

Тема 8. Делегати C#

Стисла анотація: Визначення делегатів. Правила використання делегатів. Делегати, як засоби програмування подій мовою C#. Правил оголошення делегатів і створення екземплярів делегатів.

Теми лекцій і практичних занять: «Делегати C#».

Самостійна робота здобувача освіти: опрацювання матеріалу лекцій, виконання індивідуальних завдань з використанням делегатів мовою C# відповідно варіанту, підготовка до захисту лабораторної роботи.

Тема 9. Події C#

Стисла анотація: Визначення події Windows. Правила використання подій. Утворення методів обробки подій. Використання майстрів інтегрованого середовища для програмування функцій-обробників подій. Стандартні події Windows. Події, визначені користувачем.

Теми лекцій і практичних занять: «Події C#».

Самостійна робота здобувача освіти: опрацювання матеріалу лекцій, виконання індивідуальних завдань з використанням подій мовою C# відповідно варіанту, підготовка до захисту лабораторної роботи.

Тема 10. Інтерфейси C#

Стисла анотація: Розробка консольних застосунків мовою Visual C# для обчислювання виразів різного типу із застосуванням інтерфейсного

програмування. Створення ієрархій інтерфейсів, що реалізують обробку виразів. Програмування класів, які реалізують інтерфейси. Створення об'єктів класу, який підтримує інтерфейси. Використання об'єктів інтерфейсів за допомогою оператора `is`. Використання об'єктів інтерфейсів із допомогою оператора `as`.

Теми лекції і практичних занять: «Інтерфейси C#».

Самостійна робота здобувача освіти: опрацювання матеріалу лекцій, виконання індивідуальних завдань з використанням інтерфейсів мовою C# відповідно варіанту, підготовка до захисту контрольної роботи, підготовка до модульної і контрольної робіт.

Модульний контроль 2

5. Індивідуальні завдання

Робота контрольна (РК) «Програмування на основі інтерфейсів».

1. Розробити консольну програму мовою Visual C# для обчислювання виразів різного типу із застосуванням інтерфейсного програмування. Створити ієрархію інтерфейсів, що реалізують обробку виразів, причому

- інтерфейс `IRead` визначає введення вихідних даних;
- інтерфейс `IReadWrite` розширює інтерфейс `IRead` можливістю виведення результатів на екран;
- інтерфейс `IArithm` визначає розрахунок арифметичного виразу;
- інтерфейс `IBool` визначає розрахунок логічного виразу;
- інтерфейс `IExpression` комбінує функціональність інтерфейсів і додає можливість розрахунку рядкового виразу;

клас ***CExpression***

- реалізує інтерфейси `IReadWrite`, `IExpression`;
- має конструктор без параметрів;
- реалізує відкриту функцію строкового представлення об'єкту класу

CExpression.

2. Створити об'єкт класу `CExpression` для заданих вихідних даних. Використовуючи об'єкт класу ввести вихідні дані.

3. Використовуючи об'єкти інтерфейсів `IArithm`, `IBool` і `IExpression`, за допомогою оператора `is` обчислити арифметичне, логічне і строкове вираження. Використовуючи об'єкт інтерфейсу `IReadWrite`, за допомогою оператора `as` вивести результати розрахунків.

6. Методи навчання

1. Пояснювально-ілюстративний (інформаційно-рецептивний) метод: та метод проблемного виконання (лекція).

2. Репродуктивний (практичні роботи).

3. Частково-пошуковий (евристичний) та дослідницький: (самостійна робота).

4. Дисципліна «Об'єктно-орієнтоване програмування (мова C#)» передбачає лекційні (у т. ч. з використанням мультимедійного обладнання) і практичні заняття під керівництвом викладача та самостійну роботу студента за підручниками і матеріалами, опублікованими кафедрою (методичні посібники та мережеві ресурси), що забезпечує закріплення теоретичних знань, сприяє набуттю практичних навичок і розвитку самостійного наукового мислення. Передбачено регулярні індивідуальні консультації.

7. Методи контролю

Визначення рівня засвоєння студентом навчального матеріалу дисципліни здійснюється шляхом проведення поточних (захист практичних робіт) і підсумкових контролів (захист змістовного модуля, контрольної роботи, іспит).

У завдання поточного контролю входить систематична перевірка розуміння й засвоєння студентом програмного матеріалу, виконання практичних робіт, умінь самостійно опрацьовувати тексти програм розрахунку, написання звітів, здатності усно або письмово представляти певний матеріал дослідження.

Перед підсумковим контролем ставиться завдання перевірки глибини засвоєння студентом програмного матеріалу дисципліни, логіки й взаємозв'язки між її окремими розділами, здатності творчо використовувати набуті знання, умінь сформулювати своє відношення до проблеми, що впливає зі змісту навчальної дисципліни.

8. Критерії оцінювання та розподіл балів, які отримують здобувачі освіти

Таблиця 8.1 – Розподіл балів, які отримують здобувачі освіти

Складові навчальної роботи	Бали за одне заняття (завдання)	Кількість занять (завдань)	Сумарна кількість балів
Змістовний модуль 1			
Виконання і захист практичних робіт	0...5	5	0...25
Модульний контроль	0...10	1	0...10
Змістовний модуль 2			
Виконання і захист практичних робіт	0...5	9	0...45
Виконання і захист контрольної роботи	0...5	1	0...10
Модульний контроль	0...10	1	0...10
Усього за семестр			0...100

Семестровий контроль (*іспит*) проводиться у разі відмови здобувача освіти від балів підсумкового контролю й за наявності допуску до *іспиту*. Під

час складання семестрового іспиту здобувач освіти має можливість отримати максимум 100 балів.

Білет для іспиту складається з одного теоретичного і двох практичних питань. У першому запитанні студент має продемонструвати теоретичні знання. У другому запитанні – показати навички складання і виконання комп’ютерного додатку на основі принципів структурного програмування із застосуванням функцій і файлів. У третьому пункті – продемонструвати навички зі створення та виконання застосунку з використанням класів і графічного інтерфейсу мовою програмування C# в інтегрованому середовищі Visual Studio. Максимальна кількість балів за кожне запитання та їх суми показані в табл. 8.2.

Таблиця 8.2 – Шкали оцінювання відповідей на запитання білету

Складові білету	Складові оцінки	Бали за одне питання	Сумарна кількість балів
Пункт 1. Теоретичне запитання	Надано визначення поняття	0...5	0...20
	Наведено приклад використання	0...5	
	Наведено фрагмент коду	0...10	
Пункт 2. Практичне запитання	Складено алгоритм розрахунку (блок-схема)	0...10	0...30
	Створено проект консольного додатку	0...10	
	Отримано правильні результати у консольному вікні	0...10	
Пункт 3. Практичне запитання	Складено блок-схему розрахунку	0...15	0...50
	Створено проект застосунку з графічним інтерфейсом	0...20	
	Отримано правильні результати у терміналі або формі	0...15	
Підсумкова оцінка за іспит			0...100

Таблиця 8.3 – 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 освіти протягом семестру

Задовільно (60-74) – мати знання і уміння для забезпечення програмних результатів навчання. Виконати та здати практичні роботи. Написати модульні

роботи. Уміти створювати проекти консольного типу в інтегрованому середовищі розробки програмного забезпечення Visual Studio мовою C# для проведення обчислень. Знати основи реалізації типових завдань обробки інформації.

Добре (75-89) – мати знання і уміння для забезпечення програмних результатів навчання. Виконати та здати практичні роботи. Написати модульні роботи. Додатково до вимог, які визначено для отримання задовільної оцінки: уміти створювати алгоритм розрахунку у вигляді блок-схеми і каркас класу з допомогою діаграми класів; уміти виконувати завдання у вигляді консольного застосунку але з використанням класів і інтерфейсів; уміти реалізовувати виконання варіантів специфічних умов завдань, які відрізняються від типових.

Відмінно (90-100) – мати знання, уміння й навички, що дадуть змогу самостійно, вільно та обґрунтовано відповідати на будь-які питання щодо методології та реалізації необхідних розрахунків з практики підприємств. Досконало виконати та успішно здати практичні роботи. Успішно написати модульні роботи. Уміти самостійно використовувати інтегроване середовище Visual Studio для розробки програмного забезпечення мовою C# для обробки інформації. Досконало вміти використовувати графічний інтерфейс і візуалізацію графічних результатів під час виконання завдань з обробки інформації. Уміти використовувати інструменти середовища розробки програмного забезпечення для відлагодження кодів.

9. Політика навчального курсу

Відвідування занять. Регуляція пропусків. Інтерактивний характер курсу передбачає обов'язкове відвідування практичних занять. Здобувачі освіти, які за певних обставин не можуть відвідувати практичні заняття регулярно, мають протягом тижня узгодити із викладачем графік індивідуального відпрацювання пропущених занять. Окремі пропущені заняття мають бути відпрацьовані на найближчій консультації протягом тижня після їх пропуску. Відпрацювання занять здійснюється усно у формі співбесіди за питаннями, визначеними планом заняття. В окремих випадках дозволяється письмове відпрацювання пропущених занять шляхом виконання індивідуального письмового завдання.

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 роботи здобувачів освіти будуть їх оригінальними дослідженнями або міркуваннями. Відсутність посилань на використані джерела, фабрикавання

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 плагіату 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

1. Object oriented programming using C# / Y.O. Skob, V. O. Khalturin. – Laboratory course study guide. – Kharkiv : KhAI, 2020. – 109 p. Режим доступу: http://library.khai.edu/library/fulltexts/doc/object_oriented_programming_using.pdf.

2. Основи програмування мовою C# [Електронний ресурс] : навч. посіб. до лаб. практи. / Ю. О. Скоб, М. Л. Угрюмов, В. О. Халтурін. – Х. : Нац. аерокосм. ун-т ім. М. Є. Жуковського «Харьк. авиац. ин-т», 2017. – 108 с. Режим доступу: http://library.khai.edu/library/fulltexts/metod/Skob_Osnovi_Programuvannya_movoyu_C#.pdf.

3. Programming and numerical methods. Part 2: Numerical methods using MatLab and Mathcad : guidance manual for laboratory works / O. V. Yarova, D. I. Chumachenko. – Kharkiv : National Aerospace University «KhAI», 2016. – 93 p. Режим доступу: http://library.khai.edu/library/fulltexts/metod/Yarova_Program_Method.pdf.

4. Основи програмування мовою C++. Вступ до ООП / К. П. Коробчинський, І. В. Москович, Ю.О. Скоб, О. С. Пічугіна. – Навч. посібник до лаб. практи. – Харків: Нац. аерокосм. ун-т «Харк. авіац. ін-т», 2024. – 124 с. Режим доступу: http://library.khai.edu/library/fulltexts/metod/Osnovy_Prohramuvannya_Movoyu_C++.pdf.

5. Паралельні та розподілені обчислення / Ю.О. Скоб, В. О. Халтурін. – Навч. посібник до лаб. практи. – Харків: Нац. аерокосм. ун-т «Харк. авіац. ін-т», 2023. – 116 с. Режим доступу: http://library.khai.edu/library/fulltexts/metod/Paralelni_Ta_Rozpodileni_Obchyslennya.pdf.

6. Основи програмування сучасним Фортраном / Ю.О. Скоб, М. Л. Угрюмов, В. О. Халтурін. – Навч. посібник до лаб. практи. – Харків: Нац. аерокосм. ун-т «Харк. авіац. ін-т», 2016. – 96 с. Режим доступу: http://library.khai.edu/library/fulltexts/metod/Skob_Osnovi_Programuvanna.pdf.

Посилання на курс у системі дистанційного навчання Ментор: <https://mentor.khai.edu/course/view.php?id=3139>.

11. Рекомендована література

Базова

1. Коноваленко І. В., Марущак П. О. Платформа .NET та мова програмування C# 8.0 : навч. посібник. – Тернопіль : ФОП Паляниця В. А., 2020. – 320 с. – ISBN 978-617-7875-08-5.
2. Базурін В. М., Чашечникова О. С. Об'єктно-орієнтоване програмування на мові C# : поглиблений рівень. Лабораторний практикум. – Суми : СумДУ, 2021. – 81 с.
3. Поліщук В. О. Обчислювальна техніка та програмування (розділи з C# / .NET) : навч. матеріали. – 2022 (електронне видання).
4. Парфьонов Ю. Є., Щербаков О. В. Основи об'єктно-орієнтованого програмування (матеріали для студентів, приклади на C#). – 2021 (навчальний посібник/електронний ресурс).
5. Закон України «Про вищу освіту» від 01.07.2014 р. № 1556-VII // *Відомості Верховної Ради України*. – 2014. – № 37–38. – Ст. 2004. Режим доступу: <https://zakon.rada.gov.ua/laws/show/1556-18>
6. Постанова Кабінету Міністрів України «Про затвердження Положення про акредитацію освітніх програм, за якими здійснюється підготовка здобувачів вищої освіти» від 29.12.2021 р. № 1324. Режим доступу: <https://zakon.rada.gov.ua/go/z1608-21>
7. Міністерство освіти і науки України. *Методичні рекомендації щодо розроблення стандартів вищої освіти*. – Наказ МОН України від 01.06.2024 р. № 272. Режим доступу: <https://zakon.rada.gov.ua/go/v0022729-24>
8. Національне агентство із забезпечення якості вищої освіти. *Роз'яснення щодо застосування Критеріїв оцінювання якості освітньої програми*. – Київ, 2024. Режим доступу: <https://naqa.gov.ua>
9. Міністерство освіти і науки України. *Затверджені стандарти вищої освіти: офіційний перелік*. – Київ: МОН, 2024. Режим доступу: <https://mon.gov.ua/osvita-2/vishcha-osvita-ta-osvita-doroslikh/naukovo-metodichna-rada-ministerstva-osviti-i-nauki-ukraini/zatverdzheni-standarti-vishchoi-osviti>
10. Державна служба якості освіти України. *Методичні рекомендації щодо розроблення освітньо-професійних програм фахової передвищої освіти*. – Київ, 2022. Режим доступу: https://sqe.gov.ua/wp-content/uploads/2022/06/Metodichni_rekomendacii_rozroblennya_OOP_FPO_2022.pdf
11. Національна академія наук України, Придніпровська державна академія будівництва та архітектури. *Проект стандарту освітньо-професійної програми (ОП-03-2020)*. – Дніпро, 2020. Режим доступу: <https://pgasa.dp.ua/wp-content/uploads/2020/04/proekt-standartu-OP.pdf>

Допоміжна

1. Дібрівний О. А., Гребенюк В. В. Вступ до об'єктно-орієнтованого програмування (C#) : навч. посібник. – Київ : Державний університет телекомунікацій, 2018. – 190 с.

2. Коноваленко І. В., Савків В. Б., Марущак П. О. Програмування мовою C# 7.0 : навч. посібник. – Тернопіль : ТНТУ ім. І. Пулюя, 2017. – 300 с. – ISBN 978-966-305-085-0.

3. Бублик В. В. Об'єктно-орієнтоване програмування : підручник / В. В. Бублик. – К. : ІТкнига, 2015. – 624 с.

4. Albahari J., Albahari B. *C# 9.0 in a Nutshell: The Definitive Reference*. – O'Reilly Media, 2021. – 832 p.

5. Troelsen A., Japikse P. *Pro C# 9 with .NET 5: Foundational Principles and Practices in Programming*. – Apress, April 2021. – 1383 p.

6. Price M. J. *C# 10 and .NET 6 – Modern Cross-Platform Development: Build apps, websites, and services with ASP.NET Core 6, Blazor, and EF Core 6*. – Packt / Manning (2022). – 900+ p.

7. Seemann M., van Deursen S. *Dependency Injection Principles, Practices, and Patterns*. – Manning Publications, 2020. – (covers DI patterns and practical C#/.NET examples).

8. Freeman A. *Pro ASP.NET Core 6: Develop Cloud-Ready Web Applications Using MVC, Blazor, and Razor Pages (9th ed.)*. – Apress, 2022.

12. Інформаційні ресурси

1. Microsoft Learn – C# Documentation. Офіційна документація мови C# та платформи .NET з прикладами коду, інтерактивними вправами та туторіалами. Режим доступу: <https://learn.microsoft.com/en-us/dotnet/csharp/>

2. .NET Documentation (Microsoft). Повна онлайн-документація екосистеми .NET, включаючи бібліотеки, API, ASP.NET, Entity Framework, Blazor тощо. Режим доступу: <https://learn.microsoft.com/en-us/dotnet/>

3. C# Guide – W3Schools. Інтерактивний онлайн-посібник з мови C# з прикладами, тестами та практичними завданнями. Режим доступу: <https://www.w3schools.com/cs/index.php>

4. Stack Overflow – C# Tag Професійна спільнота програмістів із сотнями тисяч запитань і рішень, пов'язаних із C#, .NET та об'єктно-орієнтованим програмуванням. Режим доступу: <https://stackoverflow.com/questions/tagged/c%23>

5. GitHub – C# Examples & Projects Repository. Відкрите сховище прикладів, навчальних проєктів і шаблонів з програмування мовою C# для різних платформ (.NET, Unity, WPF, ASP.NET). Режим доступу: <https://github.com/search?q=c%23+examples>

6. <https://w3schoolsua.github.io/cs/index.html> – український переклад W3Schools C#.