

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

Кафедра математичного моделювання та штучного інтелекту (№ 304)

ЗАТВЕРДЖУЮ

Гарант освітньої програми

Гарант освітньої програми

 Дмитро ЧУМАЧЕНКО
(підпис) (ім'я та ПРИЗВИЩЕ)

« 21 » 08 2025 р.

**СИЛАБУС ОBOB'ЯЗKОВОЇ
НАВЧАЛЬНОЇ ДИСЦИПЛІНИ**

Розподілені інформаційно-аналітичні системи
(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 ПРІЗВИЩЕ)

Погоджено з представником здобувачів освіти:

Здобувач першого (бакалаврського)

рівня вищої освіти за ОП

«Інтелектуальні системи та технології»


(підпис)

Катерина ПИЛИПЕНКО
(ім'я та ПРІЗВИЩЕ)

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	8-й
Мова викладання	Українська
Тип дисципліни	Обов'язкова
Обсяг дисципліни: кредити ЄКТС/ кількість годин	5,0 кредитів ЄКТС / 135 годин (48 аудиторних, з яких: лекції – 24, лабораторні – 24; СРЗ – 102);
Види навчальної діяльності	Лекції, лабораторні заняття, самостійна робота
Види контролю	Поточний контроль, модульний контроль, семестровий контроль – іспит
Пререквізити	«Організація та обробка електронної інформації», «Основи програмування (мова C++)», «Теорія алгоритмів і математична логіка», «Методи обчислень», «Об'єктно-орієнтоване програмування (мова C#)», «Операційні системи», «Web-програмування (.Net Framework)», «Enterprise додатки (мова Java)», «Бази даних та інформаційні системи», «Системне програмування», «Проектування програмного забезпечення», «Інтелектуальний аналіз даних», «Безпека інформаційних систем»

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

Мета – формування у здобувачів вищої освіти знань і практичних навичок про принципи побудови, проєктування, розгортання та експлуатації розподілених інформаційно-аналітичних систем, здатних ефективно обробляти великі обсяги даних у мережевому середовищі. Дисципліна спрямована на розвиток у здобувачів здатності аналізувати архітектури та технології розподілених систем, застосовувати сучасні методи оброблення, зберігання та інтеграції даних, використовувати хмарні, кластерні та сервісно-орієнтовані підходи для створення масштабованих і надійних інформаційно-аналітичних рішень.

Завдання – поетапне формування у здобувачів вищої освіти теоретичних знань про архітектурні моделі, принципи побудови, компоненти та інфраструктуру розподілених інформаційно-аналітичних систем, опанування методів і технологій збору, передачі, зберігання та оброблення даних у розподіленому середовищі, вивчення сучасних підходів до реалізації аналітичних обчислень і сервісів на базі хмарних, кластерних та сервісно-орієнтованих архітектур, набуття практичних навичок розроблення компонентів розподілених систем, взаємодії між сервісами, забезпечення узгодженості даних і відмовостійкості, ознайомлення з технологіями та інструментами для створення масштабованих аналітичних рішень, формування компетентностей у сфері побудови розподілених сховищ даних, потокової аналітики та інтеграції систем у корпоративному середовищі, розвиток умінь аналізувати ефективність розподілених обчислень, виявляти потенційні вузькі місця системи та оптимізувати архітектурні рішення.

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);
- демонструвати знання та розуміння предметної області та розуміння професійної діяльності (ЗК 3);
- спілкуватися державною мовою як усно, так і письмово (ЗК 4);
- спілкуватися іноземною мовою (ЗК 5).

Спеціальні компетентності (ФК)

Після закінчення цієї програми здобувач освіти буде здатен:

- до математичного формулювання та досліджування неперервних та дискретних математичних моделей, обґрунтовування вибору методів і підходів для розв’язування теоретичних і прикладних задач у галузі комп’ютерних наук, аналізу та інтерпретування (СК 1);
- до виявлення статистичних закономірностей недетермінованих явищ, застосування методів обчислювального інтелекту, зокрема статистичної, нейромережевої та нечіткої обробки даних, методів машинного навчання та генетичного програмування тощо (СК 2);
- до логічного мислення, побудови логічних висновків, використання формальних мов і моделей алгоритмічних обчислень, проєктування, 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);
- використовувати сучасні методи математичного моделювання об’єктів, процесів і явищ, розробляти моделі й алгоритми чисельного розв’язування задач математичного моделювання, враховувати похибки наближеного чисельного розв’язування професійних задач (СК 4);
- до системного мислення, застосування методології системного аналізу для дослідження складних проблем різної природи, методів формалізації та розв’язування системних задач, що мають суперечливі цілі, невизначеності та ризики (СК 6);
- застосовувати теоретичні та практичні основи методології та технології моделювання для дослідження характеристик і поведінки складних об’єктів і систем, проводити обчислювальні експерименти з обробкою й аналізом результатів (СК 7);
- реалізувати багаторівневу обчислювальну модель на основі архітектури клієнт-сервер, включаючи бази даних, знань і сховища даних, виконувати розподілену обробку великих наборів даних на кластерах стандартних серверів для забезпечення обчислювальних потреб користувачів, у тому числі на хмарних сервісах (СК 9);
- до інтелектуального аналізу даних на основі методів обчислювального інтелекту включно з великими та погано структурованими даними, їхньої оперативної обробки та візуалізації результатів аналізу в процесі розв’язування прикладних задач (СК 11);
- забезпечити організацію обчислювальних процесів в інформаційних системах різного призначення з урахуванням архітектури, конфігурування,

показників результативності функціонування операційних систем і системного програмного забезпечення (СК 12);

- реалізовувати високопродуктивні обчислення на основі хмарних сервісів і технологій, паралельних і розподілених обчислень при розробці й експлуатації розподілених систем паралельної обробки інформації (СК 16);

- розробляти інтелектуальне програмне забезпечення, використовувати сучасні методи штучного інтелекту (СК 17).

Програмні результати навчання (ПРН):

- Застосовувати знання основних форм і законів абстрактно-логічного мислення, основ методології наукового пізнання, форм і методів вилучення, аналізу, обробки та синтезу інформації в предметній області комп'ютерних наук (ПРН 1);

- Використовувати сучасний математичний апарат неперервного та дискретного аналізу, лінійної алгебри, аналітичної геометрії, в професійній діяльності для розв'язання задач теоретичного та прикладного характеру в процесі проєктування та реалізації об'єктів інформатизації (ПРН 2);

- Використовувати знання закономірностей випадкових явищ, їх властивостей та операцій над ними, моделей випадкових процесів та сучасних програмних середовищ для розв'язування задач статистичної обробки даних і побудови прогнозних моделей (ПРН 3);

- Використовувати методи обчислювального інтелекту, машинного навчання, нейромережевої та нечіткої обробки даних, генетичного та еволюційного програмування для розв'язання задач розпізнавання, прогнозування, класифікації, ідентифікації об'єктів керування тощо (ПРН 4);

- Використовувати методи чисельного диференціювання та інтегрування функцій, розв'язання звичайних диференціальних та інтегральних рівнянь, особливостей чисельних методів та можливостей їх адаптації до інженерних задач, мати навички програмної реалізації чисельних методів (ПРН 6);

- Розуміти принципи моделювання організаційно-технічних систем і операцій, використовувати методи дослідження операцій, розв'язання одно- та багатокритеріальних оптимізаційних задач лінійного, цілочисельного, нелінійного, стохастичного програмування (ПРН 7);

- 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);

- 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);

– Застосовувати методи та алгоритми обчислювального інтелекту та інтелектуального аналізу даних в задачах класифікації, прогнозування, кластерного аналізу, пошуку асоціативних правил з використанням програмних інструментів підтримки багатовимірної аналізу даних на основі технології Data Mining, Text Mining, Web Mining (ПРН 12);

– Виконувати паралельні та розподілені обчислення, застосовувати чисельні методи та алгоритми для паралельних структур, мови паралельного програмування при розробці та експлуатації паралельного та розподіленого програмного забезпечення (ПРН 16);

– Використовувати сучасні підходи та методи штучного інтелекту (ПРН 17).

4. Зміст навчальної дисципліни

МОДУЛЬ

Змістовний модуль 1. Загальні аспекти проектування розподілених систем

Тема 1. Предмет вивчення і задачі навчальної дисципліни

Стисла анотація: Предмет вивчення і задачі дисципліни «Розподілені інформаційно-аналітичні системи», обґрунтування потреби її вивчення. Традиційний і системний підходи до навчання комп'ютерній науці. Значення мережових та розподілених систем у сучасній обчислювальній техніці – коротка історична перспектива. Вступ у розподілені системи. Переваги та проблеми розподілених систем. Природа розподілення. Види архітектури програмного забезпечення для розподілених програм. Показники для вимірювання якості розподілених систем та програм. Ознайомлення з прозорістю доступу, розташування, тиражування, паралелізму, міграції, відмови, масштабування, продуктивності, розподілення, реалізації.

Теми лекцій і практичних занять: «Міжпроцесний однонаправлений зв'язок на основі сокетів (C++)».

Самостійна робота здобувача освіти: опрацювання матеріалу лекції, інсталяція інтегрованих середовищ Visual Studio та NetBeans, виконання індивідуальних завдань по застосуванню додатків з міжпроцесним однонаправленим зв'язком на основі сокетів мовою C++ у середовищі Visual Studio відповідно умов варіанту, підготовка до захисту лабораторної роботи.

Модульний контроль 1

Змістовний модуль 2. Процесні та комунікаційні аспекти розподілення

Тема 1. Розподілення з точки зору процесів

Стисла анотація: Базові концепції процесів. Створення процесу. Планування процесів. Планування для систем реального часу. Специфічні

алгоритми та варіанти планування, що використовуються в сучасних операційних системах. Міжпроцесна комунікація. Потоки: вступ. Інші ролі операційної системи. Використання таймерів у програмах. Прозорість з точки зору процесу. Приклад з точки зору процесу.

Теми лекції і лабораторного заняття: «Міжпроцесний однонаправлений зв'язок на основі сокетів (C#)».

Самостійна робота здобувача освіти: опрацювання матеріалу лекції, виконання індивідуального завдання по застосуванню додатків з міжпроцесним однонаправленим зв'язком на основі сокетів мовою C# у середовищі Visual Studio відповідно умов варіанту, підготовка до захисту лабораторної роботи.

Тема 2. Розподілення з точки зору комунікації

Стисла анотація: Основи комунікації. Комунікаційні технології. Односторонній зв'язок. Комунікація запит-відповідь. Двостороння передача даних. Методології адресації. Віддалений виклик процедури. Віддалений виклик методу. Інтерфейси Java. Багаторівневі моделі комунікації. Модель OSI. Модель TCP/IP. Пакет TCP/IP. Поняття IP, TCP, UDP. Адреси. Плоска проти ієрархічної адресації. Адреси на рівні посилай. Адреси на мережевому рівні. Адреси на транспортному рівні (порти). Відомі порти. Сокети. API сокета: огляд. API сокета: послідовність примітивів UDP. API сокета: послідовність примітивів TCP. Прив'язка (процесу до порту). Блокуюча та неблокуюча поведінка сокетів. Обробка неблокуючої поведінки сокета. Комунікаційний тупик. Виявлення та виправлення помилок. Специфічні протоколи програми. Інтеграція зв'язку з бізнес-логікою. Техніки, що полегшують визначення місцезнаходження компонентів один одного. Вимоги до прозорості з точки зору комунікації. Логічний і фізичний вигляди систем. Приклад із комунікаційної точки зору.

Теми лекцій і практичних занять: «Міжпроцесний однонаправлений зв'язок на основі сокетів (Java)».

Самостійна робота здобувача освіти: опрацювання матеріалу лекцій, виконання індивідуальних завдань по застосуванню додатків з міжпроцесним однонаправленим зв'язком на основі сокетів мовою Java у середовищі NetBeans відповідно умов варіанту, підготовка до захисту лабораторної роботи.

Модульний контроль 2

Змістовний модуль 3. Ресурсні й архітектурні, аспекти розподілення

Тема 1. Розподілення з точки зору ресурсів

Стисла анотація: Центральний процесор як ресурс. Пам'ять як ресурс для комунікації. Ієрархія пам'яті. Управління пам'яттю. Віртуальна пам'ять. Управління ресурсами. Статичне і динамічне виділення приватних ресурсів пам'яті. Спільні ресурси. Транзакції. Замки. Тупикова ситуація. Тиражування ресурсів. Мережа як ресурс. Пропускна здатність мережі. Методи стиснення

даних. Формат повідомлення. Сериалізація. Мережа як низка посилань. Маршрутизатори та маршрутизація. Накладні витрати на зв'язок. Механізми відновлення та їх взаємодія з перевантаженням мережі. Віртуальні ресурси. Сокети. Порти. Адреси мережі. Імена ресурсів. Дизайн розподіленої програми впливає на ефективність мережі. Прозорість з точки зору ресурсу (прозорість доступу, локації, реплікації, паралелізму, масштабування та продуктивності). Практичний приклад з точки зору ресурсів.

Теми лекції і лабораторного заняття: «Двонаправлений зв'язок між процесами на основі сокетів (C++)».

Самостійна робота здобувача освіти: опрацювання матеріалу лекцій, виконання індивідуального завдання по застосуванню додатків з міжпроцесним двонаправленим зв'язком на основі сокетів мовою C++ у середовищі Visual Studio відповідно умов варіанту, підготовка до захисту лабораторної роботи, підготовка до модульного контролю.

Тема 2. Розподілення з точки зору архітектури

Стисла анотація: Обґрунтування та огляд. Мережа та розподілення. Складність у розподілених системах. Багаторівневі архітектури. Ієрархічні архітектури. Неоднорідність: визначення та джерела. Неоднорідність продуктивності. Платформна неоднорідність. Неоднорідність операційних систем. Вплив різномірності. Портування програмного забезпечення. Архітектури апаратного та системного рівня. Тісно пов'язані (апаратні) системи. Слабозв'язані (апаратні) системи. Паралельна обробка. Архітектури програмного забезпечення. Зв'язок між програмними компонентами. Таксономія класів архітектури програмного забезпечення. Однорівневі програми. Дворівневі програми. Трирівневі програми. Багаторівневі програми. Клієнт-Сервер. Тривалість життя клієнта та сервера. Активна і пасивна сторони зв'язку. Архітектурна модель CS. Варіанти моделі CS. Сервіси зі збереженням стану та без збереження стану. Модульні та ієрархічні системи CS. Трирівнева та багаторівнева архітектури. Однорангові (рівний-до-рівного) програми. Характеристики однорангових додатків. Складність однорангового підключення. Вивчення поведінки однорангових додатків. Розподілені об'єкти. Проміжне програмне забезпечення: підтримка програмних архітектур. Робота проміжного ПЗ, огляд. Системні моделі колективних ресурсів і забезпечення обчислювальними ресурсами. Кластери. Решітки. Центри обробки даних. Хмари. Бібліотеки програмного забезпечення. Приклад бібліотеки програмного забезпечення. Статичне зв'язування та динамічне зв'язування. Ознака, що залежить від мови: файл заголовка C/C++. Апаратна віртуалізація. Віртуальні машини. Віртуальна машина Java. Статичні та динамічні конфігурації. Усвідомлення контексту. Нефункціональні вимоги до розподілених програм. Тиражування. Семантика реплікації. Реалізація реплікації. Відносини між розподіленими програмами та мережами. Прозорість з точки зору архітектури. Приклад з архітектурної точки зору. Дизайн сервера з підтримкою стану. Відокремлення завдань щодо

компонентів гри. Фізична та логічна архітектури ігрового додатку. Аспекти прозорості гри.

Теми лекцій і практичних занять: «Двонаправлений зв'язок між процесами на основі сокетів (C#)», «Двонаправлений зв'язок між процесами на основі сокетів (Java)».

Самостійна робота здобувача освіти: опрацювання матеріалу лекцій, виконання індивідуальних завдань по застосуванню додатків з міжпроцесним двонаправленим зв'язком на основі сокетів мовами C# і Java у середовищах Visual Studio і NetBeans відповідно умов варіанту, підготовка до захисту лабораторної роботи.

Модульний контроль 3

Змістовний модуль 4. Сервісне обслуговування розподілених систем

Тема 1. Розподілені системи та сервіси

Стисла анотація: Розподілені системи: обґрунтування та огляд. Прозорість доступу, місцезнаходження, реплікації, паралельності, міграції, відмови, масштабування, продуктивності, дистрибуції, імплементації. Спільні сервіси. Служби імен. Робота служби імен. Служба довідників. Проблеми розробки та впровадження служби імен. Система доменних імен (DNS). Простір імен домену. Реалізація DNS. Сервери імен DNS: повноваження та делегування. Реплікація. Докладніше про розв'язання імен. Кешування в DNS. Вивчення розпізнавання адрес. Зворотний пошук DNS. Сервіси часу. Протокол TIME. Протокол DAYTIME. Протокол мережевого часу (NTP). Синхронізація фізичного годинника. Логічні годинники та синхронізація. Алгоритми виборів. Огляд роботи. Алгоритм виборів Bully. Кільцевий алгоритм виборів. Передвибори лідера. Дослідження за допомогою алгоритму вибору. Групові комунікації. Аспекти прозорості в груповій комунікації. Служби сповіщень. Сервіси публікації та підписи. Проміжне програмне забезпечення: механізм і робота. Приклади проміжного ПЗ і технології підтримки. Загальна архітектура посередника запитів об'єктів (CORBA) Мова визначення інтерфейсу (IDL). Розширювана мова розмітки. Нотація об'єктів JavaScript (JSON). Веб-сервіси та REST (Representational State Transfer – представницька передача стану). Простий протокол доступу до об'єктів (SOAP). Детерміновані та недетерміновані аспекти розподілених систем.

Теми лекцій і практичних занять: «Багатопотоковий IPC (C++)», «Багатопотоковий IPC (C#)», «Багатопотоковий IPC (Java)».

Самостійна робота здобувача освіти: опрацювання матеріалу лекцій, виконання індивідуальних завдань по застосуванню додатків з багатопотоковим IPC мовами C++, C# і Java у середовищах Visual Studio і NetBeans відповідно умов варіанту, підготовка до захисту лабораторної роботи, підготовка до модульного контролю.

Модульний контроль 4

Змістовний модуль 5. Узагальнений погляд на сучасні розподілені системи

Тема 1. Практичні дослідження розподілення

Стисла анотація: Обґрунтування та огляд. Знайомство з прикладами використання. Приклад №1: Клієнт служби часу (з бібліотекою). Результати навчання, пов'язані з прикладом. Аналіз вимог. Архітектура та структура коду. Відокремлення занепокоєнь. Сполучення та зв'язування між компонентами. Комунікаційні аспекти дизайну. Реалізація. Тестування. Аспекти прозорості прикладу використання. Ресурси прикладу. Приклад №2: Служба сповіщень про події. Аналіз вимог. Архітектура та структура коду. Відокремлення занепокоєнь. Сполучення та зв'язування між компонентами. Комунікаційні аспекти дизайну. Сценарій використання програми для ілюстрації використання ENS. Тестування. Аспекти прозорості прикладу використання. Ресурси прикладу. Належна практика проектування для розподілених програм. Аналіз вимог. Архітектурні аспекти. Комунікаційні аспекти. Повторне використання коду, коли з'являться можливості. Створюйте бібліотеки перевіреного та надійного коду. Аспекти тестування.

Теми лекцій і практичних занять: «Багатопотоковий IPC з двома робочими потоками (C++)», «Багатопотоковий IPC з двома робочими потоками (C#)», «Багатопотоковий IPC з двома робочими потоками (Java)».

Самостійна робота здобувача освіти: опрацювання матеріалу лекцій, виконання індивідуальних завдань по застосуванню додатків з багатопотоковим IPC з двома робочими потоками мовами C++, C# і Java у середовищах Visual Studio і NetBeans відповідно умов варіанту, підготовка до захисту контрольної роботи, , підготовка до модульного контролю.

Модульний контроль 5

5. Індивідуальні завдання – немає.

6. Методи навчання

1. Пояснювально-ілюстративний (інформаційно-рецептивний) метод: та метод проблемного виконання (лекція).
2. Репродуктивний (лабораторні роботи).
3. Частково-пошуковий (евристичний) та дослідницький: (самостійна робота).
4. Дисципліна «Розподілені інформаційно-аналітичні системи» передбачає лекційні (у т. ч. з використанням мультимедійного обладнання) і лабораторні 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 лабораторних робіт) і підсумкових контролів (захист змістовного модуля, іспит).

У завдання поточного контролю входить систематична перевірка розуміння й засвоєння студентом програмного матеріалу, виконання лабораторних 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 лабораторних робіт	0...5	1	0...5
Модульний контроль 1	0...8	1	0...8
Змістовний модуль 2			
Виконання і захист лабораторних робіт	0...5	2	0...10
Модульний контроль 2	0...8	1	0...8
Змістовний модуль 3			
Виконання і захист лабораторних робіт	0...5	3	0...15
Модульний контроль 3	0...8	1	0...8
Змістовний модуль 4			
Виконання і захист лабораторних робіт	0...5	3	0...15
Модульний контроль 4	0...8	1	0...8
Змістовний модуль 5			
Виконання і захист лабораторних робіт	0...5	3	0...15
Модульний контроль 5	0...8	1	0...8
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 запитань. У відповіді на перше запитання студент має продемонструвати теоретичні знання, у відповіді на друге запитання – показати навички складання і виконання розподіленні програми, компоненти якої розташовані на локальному комп'ютері, у відповіді на третє запитання – продемонструвати знання зі створення і застосування розподіленої програми, компоненти якої розташовані у мережі. 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 наданого поняття	0... 10	0...30
	Наведено приклад використання	0... 10	
	Наведено фрагмент коду	0...10	
Пункт 2. Практичне запитання	Складено алгоритм розрахунку (блок-схема)	0...10	0...30
	Створено проект розподіленого застосунку з використанням компонентів на локальному ПК	0...10	
	Отримано запроектовані результати розрахунку	0...10	
Пункт 3. Практичне запитання	Складено алгоритм (блок-схему) розрахунку	0... 10	0...40
	Створено проект розподіленого застосунку з компонентами в мережі	0... 10	
	Отримано і проаналізовано результати розрахунку	0...20	
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 лабораторні роботи. Написати модульні роботи. Уміти створювати проекти консольного типу з міжпроцесним однонаправленим, двонаправленим зв'язком на основі сокетів та багатонаправленим зв'язком з двома робочими потоками мовою C++ на локальному комп'ютері в інтегрованому середовищі розробки програмного забезпечення Visual Studio. Знати основи реалізації типових завдань проведення розподіленого обчислення.

Добре (75-89) – мати знання і уміння для забезпечення програмних результатів навчання. Виконати та здати лабораторні роботи. Написати модульні роботи. Додатково до вимог, які визначено для отримання задовільної оцінки: уміти створювати проекти з міжпроцесним однонаправленим, двонаправленим зв'язком на основі сокетів та багатонаправленим зв'язком з двома робочими потоками мовою C# на локальному комп'ютері у середовищі Visual Studio; 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 міжпроцесного зв'язку у розподілених додатках. Досконало виконати та успішно здати лабораторні роботи. Успішно написати модульні роботи. Уміти самостійно використовувати середовища Visual Studio і NetBeans для розробки та відлагодження розподілених додатків з міжпроцесним однонаправленим, двонаправленим зв'язком на основі сокетів та багатонаправленим зв'язком з двома робочими потоками мовами C++, C# і Java. Досконало вміти використовувати інструменти середовищ програмування для налагодження 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 лабораторні 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. Паралельні та розподілені обчислення / Ю.О. Скоб, В. О. Халтурін. – Навч. посібник до лаб. практик. – 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>.

2. 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>.

3. Основи програмування мовою C# [Електронний ресурс] : навч. посіб. до лаб. практик. / 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>.

4. 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>.

5. Основи програмування мовою C++. Вступ до ООП / К. П. Коробчинський, І. В. Москович, Ю.О. Скоб, О. С. Пічугіна. – Навч. посібник до лаб. практик. – 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.

6. Основи програмування сучасним Фортраном / Ю.О. Скоб, М. Л. Угрюмов, В. О. Халтурін. – Навч. посібник до лаб. практик. – 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=3148>.

11. Рекомендована література

Базова

1. Купін А. І., Музика І. О. Мережеві інформаційні технології : практикум. – Київ : ЦУЛ, 2018. – 236 с. – ISBN 978-966-7250-18-9.

2. Райчев І. Е., Харченко О. Г., Замковий В. В. Принципи проектування відкритих розподілених систем : навч. посіб. – Київ : НАУ, 2019. – 172 с. – Режим доступу: <https://er.nau.edu.ua/handle/NAU/30135>.

3. Тарасов Д., Андрухів А., Сокіл М. Розподілені інформаційні системи : електронний навчально-методичний комплекс. – Львів : Національний університет «Львівська політехніка», 2021. – 1 електрон. опт. диск (CD-ROM). – Режим доступу: <https://opac.lpnu.ua/bib/1135457>.

4. Шликов В. В., Данілова В. А. Високопродуктивні розподілені обчислювальні системи : практикум. – Київ : КП ім. Ігоря Сікорського, 2018. – 145 с. – Режим доступу: <https://ela.kpi.ua/items/5fc00216-4a8f-41ed-aa0f-305ca2465517>.

5. Луцків А. М., Лупенко С. А., Пасічник В. В. Паралельні та розподілені обчислення : навч. посіб. – Львів : Магнолія, 2015. – 284 с. – ISBN 978-617-574-110-8.

6. 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>

7. 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>

8. 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>

9. Національне агентство із забезпечення якості вищої освіти. *Роз'яснення щодо застосування Критеріїв оцінювання якості освітньої програми*. – Київ, 2024. Режим доступу: <https://naqa.gov.ua>

10. 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>

11. 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

12. 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. Raptis D. Distributed Systems for Practitioners. – Leanpub, 2021. – 180 с. – Режим доступу: <https://leanpub.com/distributed-systems-for-practitioners>

2. Mitropoulos S. **D**istributed Information Systems and their Management. – Ελληνικό Ανοικτό Πανεπιστήμιο, 2023. – 230 с. – Режим доступу: <https://repository.kallipos.gr/handle/11419/10301>

3. Tanenbaum A. S., Van Steen M. Distributed Systems: Principles and Paradigms. – 2nd ed. – Prentice Hall, 2007 (практичною актуалізацією). – 698 с. – Режим доступу: vowi.fsinf.at

4. Цеслів О. В., Коломієць А. С. Інформаційно-аналітичні системи та технології: комп'ютерний практикум [Ел. ресурс] / О. В. Цеслів, А. С. Коломієць. – Київ : КПІ ім. Ігоря Сікорського, 2024. – 141 с. – Режим доступу: ela.kpi.ua

5. «Розподілені та хмарні інформаційно-аналітичні системи» [Ел. курс / методичні матеріали]. – Харків : ХПІ, 2024. – Режим доступу: web.kpi.kharkov.ua

6. Захарова І. В., Філіпова Л. Я., Задорожний І. С., Тарасенко Д. А. Основи інформаційно-аналітичної діяльності : навч. посібник / І. В. Захарова та ін. – Черкаси : Східноєвропейський університет, 2-е вид., випр. і доповн., 2024. – 347 с. – Режим доступу: suem.edu.ua

12. Інформаційні ресурси

1. Microsoft Learn. Distributed systems documentation [Електронний ресурс]. – Режим доступу: <https://learn.microsoft.com/en-us/azure/architecture/guide/distributed-systems>.

2. IBM Developer. Distributed systems and microservices architecture [Електронний ресурс]. – Режим доступу: <https://developer.ibm.com/articles/intro-to-distributed-systems>.

3. Red Hat. Introduction to distributed systems [Електронний ресурс]. – Режим доступу: <https://www.redhat.com/en/topics/microservices/what-is-a-distributed-system>.

4. Coursera. Distributed Systems courses [Электронный ресурс]. – Режим доступа: <https://www.coursera.org/courses?query=distributed%20systems>.
5. GeeksforGeeks. Distributed Computing Concepts [Электронный ресурс]. – Режим доступа: <https://www.geeksforgeeks.org/distributed-computing/>.
6. Kubernetes Documentation. Designing distributed applications [Электронный ресурс]. – Режим доступа: <https://kubernetes.io/docs/concepts/architecture/>.
7. Google Cloud Architecture Center. Distributed system design patterns [Электронный ресурс]. – Режим доступа: <https://cloud.google.com/architecture>.
8. Spring Framework Documentation. Distributed computing with Spring Cloud [Электронный ресурс]. – Режим доступа: <https://spring.io/projects/spring-cloud>.
9. Stack Overflow. Distributed systems Q&A community [Электронный ресурс]. – Режим доступа: <https://stackoverflow.com/questions/tagged/distributed-systems>.
10. Wikipedia. Distributed computing [Электронный ресурс]. – Режим доступа: https://en.wikipedia.org/wiki/Distributed_computing.