

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ  
«Київський політехнічний інститут імені Ігоря Сікорського»

*Затверджую*

Голова Приймальної комісії  
Ректор



*[Handwritten signature]*

Анатолій МЕЛЬНИЧЕНКО

28.03.2025р.

дата

**Навчально-науковий інститут атомної і теплової енергетики**

повна назва факультету/навчально-наукового інституту

**ПРОГРАМА  
фахового іспиту**

для вступу на освітньо-професійну та освітньо-наукову програму підготовки магістра  
«Інженерія програмного забезпечення інтелектуальних кібер-фізичних систем в енергетиці»

*за спеціальністю F2 Інженерія програмного забезпечення*

Програму ухвалено:

Вченою Радою Навчально-наукового інституту  
атомної і теплової енергетики

Протокол № 9 від 24 лютого 2025 р.

Голова Вченої Ради

*[Handwritten signature]*

Євген ПИСЬМЕННИЙ

## ВСТУП

Програма фахового іспиту визначає форму організації, зміст та особливості проведення вступного фахового іспиту на освітньо-професійну та освітньо-наукову програму підготовки магістра «Інженерія програмного забезпечення інтелектуальних кібер-фізичних систем в енергетиці» за спеціальністю F2 Інженерія програмного забезпечення.

**Метою програми** фахового іспиту для вступу на освітньо-професійну та освітньо-наукову програму підготовки магістра «Інженерія програмного забезпечення інтелектуальних кібер-фізичних систем в енергетиці» за спеціальністю F2 Інженерія програмного забезпечення є перевірка набуття вступником компетентностей та результатів навчання, що визначені стандартом вищої освіти за спеціальністю 121 Інженерія програмного забезпечення для першого (бакалаврського) рівня вищої освіти.

## 1. ОСНОВНИЙ ВИКЛАД

### 1.1. Перелік тем, які виносяться на фаховий іспит

Повний перелік тем, які виносяться на вступний фаховий іспит для вступу за освітньо-професійною програмою (ОПП) та освітньо-науковою програмою (ОНП) «Інженерія програмного забезпечення інтелектуальних кібер-фізичних систем в енергетиці» підготовки магістрів спеціальності F2 Інженерія програмного забезпечення.

#### 1. Алгоритми та обчислювальна складність

##### 1.1. Основи структури даних і алгоритми

##### 1.1.1. Поняття алгоритму. Визначення його часової та просторової складності

##### 1.1.2. Поняття абстрактного типу даних.

Стандартні абстрактні типи даних: стеки, списки, вектори, словники, множини, мультимножини, черги, черги з пріоритетами

##### 1.1.3. Кортежі, множини, словники, одно- та двозв'язні списки.

Реалізація абстрактних типів даних з оцінюванням складності операцій

##### 1.1.4. Базові алгоритми та їх складність: пошук, сортування (прості сортування видаленням, вставками, обмінами та удосконалені сортування деревом, сортування Шела, швидке сортування)

##### 1.1.5. Алгоритми на графах та їх складність: пошук в ширину і глибину; пошук зв'язних компонентів; побудова кістякового дерева; побудова найкоротших шляхів з виділеної вершини; побудова найкоротших шляхів між двома вершинами

##### 1.2. Стратегії розроблення алгоритмів

##### 1.2.1. Стратегія «зменшуй та володарюй» та приклади застосування.

Стратегія «розділяй та володарюй» та приклади застосування.

Стратегія балансування та приклади застосування.

Динамічне програмування та приклади застосування.

Оцінювання складності алгоритму під час застосування кожної стратегії

##### 1.3. Моделі обчислень

##### 1.3.1. Імперативний та декларативний підходи до програмування

##### 1.3.2. Розв'язні, напіврозв'язні та нерозв'язні проблеми. Проблема зупинки

## **2. Архітектура обчислювальних систем**

### **2.1 Функції бінарної логіки**

### **2.2 Представлення даних на рівні машин**

2.2.1. Системи числення, двійкове, вісімкове, шістнадцяткове числення. Представлення чисел у цілочисельному форматі та форматі із плаваючою комою. Доповнювальний код

2.2.2. Основні арифметичні операції над цілочисельними двійковими числами. Формат чисел з плаваючою комою, переваги та недоліки, основні арифметичні операції та їх проблеми

### **2.3 Пристрої введення-виведення. Поняття шини комп'ютера**

### **2.4 Функціональна організація**

2.4.1 Структура комп'ютера, класична архітектура фон Неймана. Види пам'яті: кеш-пам'ять, оперативна пам'ять, зовнішня пам'ять. CPU, GPU. Периферійні пристрої

## **3. Бази даних та сховища даних**

3.1. Ключі та нормалізація даних: основні нормальні форми (1NF, 2NF, 3NF, BCNF, 4NF)

3.2. Основні концепції систем баз даних: модель даних; мова запитів; транзакція; ACID-властивості транзакції, індексування; резервне копіювання та відновлення; розподіленість та реплікація даних; безпека даних

3.3. Моделювання даних: створення моделі даних для інформаційної системи; концептуальна, логічна, фізична моделі даних; ER-модель; нотації ER-моделей

3.4. Реляційні бази даних: особливості організації та зберігання даних у реляційних базах даних; основні характеристики реляційних баз даних; DBMS (Database Management System)

3.5. Побудова запиту: мови SQL (structured query language), DDL (Data Definition Language), DML (Data Manipulation Language), DCL (Data Control Language), TCL (Transaction Control Language)

3.6. Обробка запитів: основні операції реляційної алгебри: відбір (selection), проєкція (projection), об'єднання (union), перетин (intersection), різниця (difference), декартовий добуток (cartesian product), об'єднання за атрибутом (Join), ділення (Division)

3.7. Розподілені бази даних/хмарні обчислення: доступність, масштабованість, виклики, технології

3.8. Особливості, переваги і недоліки моделей напівструктурованих і неструктурованих баз даних: моделі даних Ключ-значення (Key-Value), Документо-орієнтовані (Document-Oriented), Столпцево-орієнтовані (Column-Family), Графові (Graph), Масив-орієнтовані (Array-Based).

## **4. Інженерія систем і програмного забезпечення**

### **4.1. Складні системи**

4.1.1. Класифікація систем за призначенням, взаємодією із зовнішнім середовищем, походженням, видом елементів, способом організації

4.1.2. Складні та великі системи. Властивості та характерні особливості складних систем

4.1.3. Поняття системи та її структури

4.1.4. Поняття декомпозиції та агрегування

### **4.2. Моделі систем**

4.2.1. Моделювання систем

- 4.2.2. Зв'язок між системою та моделлю. Ізо- та гомоморфізм
- 4.2.3. Класифікація моделей систем
- 4.3. Аналіз вимог
  - 4.3.1. Джерела та методи збирання вимог
  - 4.3.2. Вимоги користувача: модель вимог на основі прецедентів (варіантів використання) (Use Case Diagram), історії користувачів (user story). Вимоги до описів варіантів використання
  - 4.3.3. Класифікація вимог до програмного забезпечення: функціональні та нефункціональні вимоги, обмеження, структуризація функціональних вимог.
- 4.4. Проєктування програмного забезпечення
  - 4.4.1. Моделювання проєкту з UML: діаграми статичні та динамічні, логічні та фізичні
  - 4.4.2. Види проєктування: архітектурне (верхній рівень) та деталізоване проєктування (класів, атрибутів, операцій), проєктування інтерфейсу користувача
  - 4.4.3. Парадигми проєктування: функціональна декомпозиція згори вниз, архітектура, орієнтована на дані, об'єктно-орієнтований аналіз та проєктування, подієво-керована архітектура
  - 4.4.4. Ідентифікація класів предметної області. UML-діаграми ієрархії класів: моделювання підсистем, класів та зв'язків між ними
  - 4.4.5. Проєктування сценаріїв реалізації варіантів використання на основі UML-діаграм послідовностей та комунікації
  - 4.4.6. Роль архітектури. Стандартні архітектури: клієнт-серверна та n-рівнева архітектура, Model View Controller
  - 4.4.7. Архітектурні моделі та патерни проєктування (Abstract Factory, Facade, Decorator, Flyweight, Visitor, Observer, Proxy, Strategy, Chain of Responsibility)
- 4.5. Реалізація програмного забезпечення
  - 4.5.1. Вимоги до оформлення коду: стиль, розбиття на структуровані одиниці, найменування змінних, класів, об'єктів тощо
  - 4.5.2. Засоби автоматичної генерації програмного коду на основі трансформацій UML-модель – код ООП-мовою, повторне використання коду ПЗ
  - 4.5.3. Налаштування: Точки зупинки (Breakpoints), Спостереження за змінними (Variable Watch), Виведення на консоль (Console Output), Налаштовувач (Debugger), Аналізатори коду (Code Analyzers)
  - 4.5.4. Керування конфігурацією програмного забезпечення та контроль версій
  - 4.5.5. Постійна інтеграція/постійне впровадження (Continuous Integration/Continuous Delivery)
- 4.6. Забезпечення якості: тестування, верифікація, валідація
  - 4.6.1. Призначення, спільне та відмінності процесів тестування, верифікації, валідації
  - 4.6.2. Види тестів: модульні, інтеграційні, регресійні, системні, валідаційні
  - 4.6.3. Тестування методами білої та чорної скрині
  - 4.6.4. Розробка через тестування (Test-driven development)
  - 4.6.5. Додаткові техніки верифікації та валідації: інспекція коду, перевірка на відповідність стандартам і вимогам, оцінювання зручності використання та користувацького досвіду, перевірка продуктивності та масштабованості
- 4.7. Командна робота, підходи до розробки програмного забезпечення (ПЗ)
  - 4.7.1. Класичні моделі розробки ПЗ: каскадно-водопадна, ітераційна, інкрементна

- 4.7.2. Промислові технології розробки. RUP- уніфікований підхід, керований варіантами використання, архітектурно-центрований, ітераційний, інкрементний
- 4.7.3. Ролі та обов'язки у програмній команді, переваги командної роботи, ризики та складність такої співпраці
- 4.7.4. Технології гнучкої розробки ПЗ та їх особливості: Agile, Scrum, Extreme Programming (XP), Kanban
- 4.7.5. Моделі керування командною роботою (на основі UML діаграм Ганта і Перта)
- 5. Кібербезпека та захист інформації**
- 5.1. Основи кібербезпеки
  - 5.1.1. Поняття кіберпростору та інформаційного простору
  - 5.1.2. Інформаційна безпека як сфера національної безпеки України
  - 5.1.3. Поняття кібербезпеки, захисту інформації та кіберзахисту
  - 5.1.4. Види захисту інформації
  - 5.1.5. Поняття конфіденційності, цілісності, доступності
  - 5.1.6. Принципи кібербезпеки
- 5.2. Кіберзагрози та кібератаки
  - 5.2.1. Поняття загроз, атак, вразливості
  - 5.2.2. Класифікація загроз, атак
  - 5.2.3. Кіберзлочини. Кібервійна. Кібероборона
  - 5.2.4. Кібертероризм. Кіберрозвідка
  - 5.2.5. Модель порушника
  - 5.2.6. Поняття, сутність та основні завдання комплексної системи захисту інформації
- 5.3. Безпека мережі
  - 5.3.1. Поняття про шкідливе програмне забезпечення
    - 5.3.1.1 Поняття про шпигунські програми, фішинг, програми-вимагачі, DDoS-атаки, соціальну інженерію
  - 5.3.2. Способи забезпечення безпеки мережі
    - 5.3.2.1 Поняття про брандмауери, контроль доступу, сегментацію мережі
    - 5.3.2.2 Системи виявлення та запобігання вторгненням
  - 5.3.3. Процедури ідентифікації, автентифікації, авторизації користувачів мережі
    - 5.3.3.1 Поняття процедур ідентифікації, автентифікації та авторизації користувача
    - 5.3.3.2 Види ідентифікації, автентифікації та авторизації користувача
- 5.4. Криптографічні методи захисту даних
  - 5.4.1. Класифікація методів криптографічного захисту за призначенням
  - 5.4.2. Сутність криптографії, криптології, стеганографії
  - 5.4.3. Поняття моделі симетричної криптосистеми
  - 5.4.4. Шифри перестановки (шифр перестановки за ключем), шифр простої заміни (шифр Цезаря), шифр складної заміни (шифр Віженера)
  - 5.4.5. Поняття симетричних блочних алгоритмів шифрування
  - 5.4.6. Поняття моделі асиметричної криптосистеми
  - 5.4.7. Поняття процесу автентифікації документів. Електронний підпис
- 6 Математика в ІТ**
- 6.1 Основний понятійний апарат математичного аналізу

- 6.1.1. Числова послідовність та її границя. Нескінченно малі та великі величини. Порівняння нескінченно малих і великих величин
- 6.1.2. Функції однієї змінної. Границя функції в точці. Неперервні функції
- 6.1.3. Похідна та її застосування для дослідження функцій однієї змінної
- 6.1.4. Невизначені, визначені інтеграли: поняття та застосування
- 6.1.5. Функції багатьох змінних. Частинні похідні. Необхідні і достатні умови екстремуму, умовного екстремуму
- 6.1.6. Метод найменших квадратів (лінійна залежність)
- 6.1.7. Числові ряди. Поняття їх збіжності
- 6.1.8. Основні означення теорії диференціальних рівнянь: порядок диференціального рівняння, частинний розв'язок, загальний розв'язок, загальний інтеграл, задача Коші
- 6.2. Елементи аналітичної геометрії
  - 6.2.1. Пряма і площина в просторі. Поняття гіперплощини
  - 6.2.2. Криві другого порядку. Еліпс, гіпербола, парабола та їх властивості
  - 6.2.3. Поняття поверхні, її типи
- 6.3. Елементи лінійної алгебри
  - 6.3.1. Матриці. Дії з матрицями. Визначники. Обернена матриця
  - 6.3.2. Власні вектори та власні числа матриці
  - 6.3.3. Системи лінійних алгебраїчних рівнянь, умови їх розв'язуваності. Методи їх розв'язання
  - 6.3.4. Лінійний векторний простір та його основні властивості. Розмірність і базис простору
- 6.4. Методи оптимізації
  - 6.4.1. Основні поняття та цілі в задачах лінійного та нелінійного програмування. Градієнтний метод: ідея та алгоритм
- 6.5. Дискретна математика
  - 6.5.1. Множини. Поняття чітких та нечітких множин. Операції над чіткими множинами: об'єднання, перетин, різниця, доповнення, булеан множини, декартів добуток
  - 6.5.2. Бінарні відношення та їх властивості: рефлексивність, симетричність, транзитивність
  - 6.5.3. Комбінаторний аналіз. Правило суми та добутку. Сполуки, перестановки, розміщення: без повторень та з повтореннями. Принцип включень і виключень
  - 6.5.4. Елементи математичної логіки. Пропозиційна логіка. Логіка висловлювань. Логічні сполучники. Атомарні формули. Таблиці істинності
  - 6.5.5. Графи. Типи графів: Орієнтовні та неорієнтовні графи. Вершини та ребра, степінь вершини, суміжність. Ізоморфізм графів. Операції над графами: об'єднання, пряма сума, доповнення, вилучення ребра, вилучення вершини
  - 6.5.6. Маршрути, ланцюги, цикли та їх різновиди у графах
  - 6.5.7. Зв'язність графів, компоненти зв'язності неорієнтованих графів. Відстань між вершинами
  - 6.5.8. Древа, ліси: основні поняття
- 6.6. Основний понятійний апарат теорії ймовірностей та математичної статистики
  - 6.6.1. Стохастичний експеримент. Простір елементарних подій. Операції над подіями. Класична, геометрична, статистична, аксіоматична ймовірність. Умовні ймовірності

- 6.6.2. Формула повної ймовірності. Формула Байєса. Схема незалежних випробувань Бернуллі. Закон великих чисел
- 6.6.3. Одновимірні дискретні випадкові величини. Числові характеристики дискретних випадкових величин. Моменти дискретних випадкових величин
- 6.6.4. Закони розподілу неперервних випадкових величин: рівномірний, нормальний
- 6.6.5. Багатовимірні дискретні величини та їх числові характеристики. Коефіцієнт кореляції
- 6.6.6. Поняття випадкової функції і процесу
- 6.6.7. Основні задачі математичної статистики. Первинна обробка даних. Числові характеристики вибіркової сукупності
- 6.6.8. Статистичний та інтервальний ряди розподілу. Гістограма
- 6.6.9. Точкові та інтервальні оцінки. Довірчі інтервали
- 6.6.10. Перевірка статистичних гіпотез. Критерії згоди. Критерій Пірсона

## **7 Мережі та обмін даними**

- 7.1. Основні поняття та функції комп'ютерних мереж
  - 7.1.1. Класифікація та функції комп'ютерних мереж. Комутація каналів та комутація пакетів. Топології комп'ютерних мереж
  - 7.1.2. Поняття протоколу та інтерфейсу, ієрархія протоколів, потік інформації в мережі. Еталонні моделі ISO/OSI та TCP/IP
- 7.2. Типи мережевих сервісів, сервіси зі встановленням з'єднань та без встановлення з'єднань. Основні функції. Поняття портів та сокетів. Протоколи TCP та UDP.
- 7.3. Маршрутизація та обмін даними
  - 7.3.1. Функції мережевого рівня, задача маршрутизації. Протокол IP. IP-адреси та їх властивості
- 7.4. Технології бездротових мереж
  - 7.4.1. Принципи організації бездротових мереж
  - 7.4.2. Технології Wi-Fi
- 7.5. Безпека мережі: на каналному рівні, VLAN, VPN. Основні захищені мережеві протоколи

## **8. Операційні системи**

- 8.1. Призначення операційних систем
  - 8.1.1. Різноманітність операційних систем (однокористувацькі, багатокористувацькі, реального часу, вбудовані системи)
  - 8.1.2. Основні функції операційних систем
  - 8.1.3. Вимоги до операційних систем, поняття відмовостійкості
- 8.2. Принципи побудови операційної системи
  - 8.2.1. Типи архітектур ядра операційної системи
  - 8.2.2. Привілейований режим і режим користувача
  - 8.2.3. Системні виклики
- 8.3. Одночасність
  - 8.3.1. Мультизадачність
  - 8.3.2. Мультипроцесорність
  - 8.3.3. Паралельність
- 8.4. Модель процесу

- 8.4.1. Блок керування процесом
- 8.4.2. Контекст процесу
- 8.4.3. Стани процесу
- 8.4.4. Розподіл пам'яті (типи адрес, методи розподілу пам'яті)
- 8.4.5. Віртуальна пам'ять (сторінкова, сегментна, сегментно-сторінкова організація пам'яті, свопінг)
- 8.5. Файлові системи
  - 8.5.1. Основні поняття про файли і файлові системи
  - 8.5.2. Логічна та фізична організація файлів
- 8.6. Поняття системи реального часу
  - 8.6.1. Визначення систем реального часу, основні характеристики
  - 8.6.2. Види систем реального часу та їх відмінності
- 9. Основи програмування**
  - 9.1. Об'єктно-орієнтоване програмування
    - 9.1.1. Поняття класу та об'єкта; конструктор і деструктор, інтерфейс та реалізація
    - 9.1.2. Базові концепції ООП: абстракція, інкапсуляція, спадкування, поліморфізм
    - 9.1.3. Зв'язки між класами: асоціація, агрегація, композиція, спадкування, залежність, реалізація
  - 9.2. Принципи та сфера застосування видів програмування: функціональне, логічне, подійне, реактивне, генеративне програмування
  - 9.3. Паралельні та розподілені обчислення
    - 9.3.1. Моделі паралельних обчислень
    - 9.3.2. Ефективність та вартість паралельних обчислень
    - 9.3.3. Закон Амдаля
    - 9.3.4. Синхронне та асинхронне програмування
  - 9.4. Трансляція та виконання
    - 9.4.1. Компілятор, інтерпретатор, компоувальник, компілятор в байт-код або проміжний код, JIT компілятор, система виконання (Runtime)
    - 9.4.2. Форма Backus-Naur (БНФ) та розширена нотація БНФ
    - 9.4.3. Регулярні вирази
- 10. Штучний інтелект**
  - 10.1. Фундаментальні поняття
    - 10.1.1. Інтелект, штучний інтелект, поняття агента і середовища, задачі штучного інтелекту, раціональність, сильний і слабкий штучний інтелект, ризики штучного інтелекту
  - 10.2. Пошук у просторі станів
    - 10.2.1. Стратегії пошуку: пошук в ширину, пошук в глибину, двонаправлений пошук, жадібний алгоритм.
  - 10.3. Основи подання знань
    - 10.3.1. Факти, знання, властивості знань. Моделі знань: семантичні мережі, фрейми, логічні моделі, продукційні правила
  - 10.4. Машинне навчання
    - 10.4.1. Навчання з вчителем та без, навчання з підкріпленням, регресійні і класифікаційні задачі
    - 10.4.2. Лінійна і логістична регресія: ідентифікація, регуляризація, сфера застосування



10.4.3. Поняття: формальний нейрон, штучна нейронна мережа, функції активації формального нейрона

## 1.2. Порядок проведення фахового іспиту

Фаховий іспит проводиться у вигляді письмової роботи. Кожний екзаменаційний білет фахового іспиту містить п'ять завдань. До екзаменаційного білету фахового іспиту включаються як теоретичні так і практичні завдання. Для фахового іспиту передбачено 30 екзаменаційних білетів, сформованих з наведеного вище переліку тем.

Термін виконання фахового іспиту становить 4 академічні години (180 хвилин) без перерви. Після написання роботи атестаційна комісія перевіряє її та виставляє оцінку згідно з критеріями оцінювання.

Методика проведення фахового іспиту наступна. Члени атестаційної комісії інформують вступників про порядок проведення та оформлення робіт з фахового іспиту видають вступникам екзаменаційні білети з відповідними варіантами та заздалегідь роздруковані підписані листи для написання робіт. Надалі в ці листи вступники записують письмові відповіді на питання екзаменаційного білету і наприкінці зазначають дату та ставлять особистий підпис.

На організаційну частину фахового іспиту (пояснення по проведенню, оформленню і критеріям оцінювання іспиту, видачі білетів і листів для написання роботи) відводиться 20 хвилин від усього часу фахового іспиту, на відповіді та розв'язок на кожне з п'яти завдань екзаменаційного білету вступнику надається по 30 хвилин і на заключну частину (збір білетів і письмових робіт у вступників членами конкурсної комісії) – 10 хвилин.

Після закінчення етапу написання фахового іспиту, проводиться перевірка відповідей та їх оцінювання всіма членами комісії. Члени атестаційної комісії приймають спільне рішення щодо виставлення оцінки на відповідь до кожного з питань екзаменаційного білету. Ці оцінки виставляються на аркуші з відповідями студента.

Підведення підсумку фахового іспиту здійснюється шляхом занесення балів в екзаменаційну відомість. Ознайомлення студента з результатами іспиту проводиться згідно з правилами прийому в університет.

## 1.3. Допоміжні матеріали для складання фахового іспиту

Під час складання фахового іспиту заборонено використання допоміжної літератури та інших допоміжних матеріалів та засобів.

## 1.4. Критерії оцінювання фахового іспиту

Критерії оцінювання завдань фахового іспиту враховують наступне:

– оцінка за виконання фахового іспиту виставляється за системою ECTS - 100-бальна шкала;

– максимальна кількість балів, яка нараховується за виконання окремого завдання 1 – 20 балів, завдання 2 – 20 балів, завдання 3 – 20 балів, завдання 4 – 20 балів, завдання 5 – 20 балів.

– оцінювання результатів кожного завдання здійснюється у п'ятирівневій системі балів (таблиця 1).

Таблиця 1. Критерії оцінювання завдань фахового іспиту

Оцінка	Опис
16-20	Наведені необхідні теоретичні відомості (основні теоретичні положення методу розв'язання задачі), повністю виконано практичне завдання
10-14	Наведені необхідні теоретичні відомості (основні теоретичні положення методу розв'язання задачі), є деякі несуттєві недоліки при виконанні практичного завдання
6-9	Наведені деякі теоретичні відомості (деякі теоретичні положення методу розв'язання задачі), повністю виконано практичне завдання
1-5	Наведені деякі теоретичні відомості (деякі теоретичні положення методу розв'язання задачі) без виконання практичного завдання
0	Теоретичні відомості відсутні, практичне завдання не виконано

Загальний критерій оцінюється по сумі балів за відповіді на завдання із всіх п'яти розділів, максимальна оцінка складає 100 балів:

$$R_{\text{заг}} = R_{\text{розд1}} + R_{\text{розд2}} + R_{\text{розд3}} + R_{\text{розд4}} + R_{\text{розд5}} = 20 + 20 + 20 + 20 + 20 = 100 \text{ балів}$$

Згідно чинних «Правил прийому до КПІ ім. Ігоря Сікорського в 2025 році» при обчисленні конкурсного балу використовується шкала оцінювання від 100 до 200 балів. Перерахунок загального показника фахового випробування у рейтингову оцінку фахового випробування здійснюється згідно наступної таблиці:

Таблиця відповідності оцінок РСО (60...100 балів) оцінкам 200-бальної шкали (100...200 балів)

шкала РСО	шкала 100...200	шкала РСО	шкала 100...200	шкала РСО	шкала 100...200	шкала РСО	шкала 100...200
60	100	70	140	80	160	90	180
61	105	71	142	81	162	91	182
62	110	72	144	82	164	92	184
63	115	73	146	83	166	93	186
64	120	74	148	84	168	94	188
65	125	75	150	85	170	95	190
66	128	76	152	86	172	96	192
67	131	77	154	87	174	97	194
68	134	78	156	88	176	98	196
69	137	79	158	89	178	99	198
						100	200

### 1.5. Приклад типового завдання фахового іспиту

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ  
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ СІКОРСЬКОГО»

Спеціальність F2 Інженерія програмного забезпечення  
Освітня програма Інженерія програмного забезпечення інтелектуальних кібер-фізичних систем в енергетиці

#### ЕКЗАМЕНАЦІЙНИЙ БІЛЕТ № 1

фахового іспиту

для вступу на освітньо-професійну програму підготовки магістра

##### Завдання 1. (має 20 балів)

Поняття, сутність та основні завдання комплексної системи захисту інформації. (В описі обов'язкове наведення прикладів).

##### Завдання 2. (має 20 балів)

Класифікація та функції комп'ютерних мереж. Комутація каналів та комутація пакетів. Топології комп'ютерних мереж. (В описі обов'язкове наведення прикладів).

##### Завдання 3. (має 20 балів)

Ідентифікація класів предметної області. UML-діаграми ієрархії класів: моделювання підсистем, класів та зв'язків між ними. (В описі обов'язкове наведення прикладів).

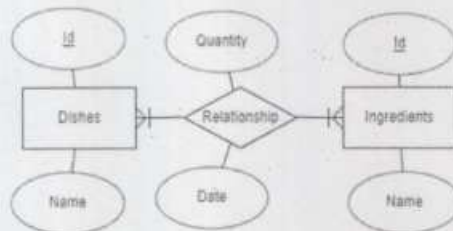
##### Завдання 4. (має 20 балів)

На мові програмування C, C++, C# або Java написати фрагмент програми (функцію, метод або клас), для розв'язання наступної задачі:

Задано одновимірні масиви A(n) та B(n). Сформувавши одновимірний масив C(2\*n), елементами якого є елементи масивів A(n) та B(n), записані через один (a1, b1, a2, b2, a3, b3, ...).

##### Завдання 5. (має 20 балів)

- 1)
- 2)
- 3)



Для наведеної ER діаграми створіть наступні скрипти та запит:

- a) DDL (CREATE, ALTER) – створення таблиць, формалізація зв'язку, внесення необхідних обмежень цілісності
- b) DML (INSERT) – додайте по 7 записів до кожної таблиці. UPDATE – оновіть записи в асоціативній таблиці, змініть дату на DateNew в тих записах, де назви інгредієнтів починаються з символів XXX (довільно за Вашим вибором).
- c) SELECT – оберіть всі інгредієнти, які були додані до блюд ДО певної дати Date1 (довільно за Вашим вибором).

Затверджено на засіданні кафедри інженерії  
програмного забезпечення в енергетиці,  
протокол № 26 від 21 лютого 2025 р.

В.о. завідувача кафедри ІІЗЕ

Олександр КОВАЛЬ

## 2. ПРИКІНЦЕВІ ПОЛОЖЕННЯ

1. Особи, які без поважних причин не з'явилися на вступні іспити у визначений розкладом час, особи, знання яких було оцінено балами нижче встановленого цим Положенням рівня, до участі в наступних вступних іспитах і в конкурсному відборі не допускаються.

2. В разі неможливості проведення іспиту в очному режимі, випробування можуть проводитися в дистанційній формі з використанням технологій дистанційного навчання «Google» та сервісу відеотелефонного зв'язку «GoogleMeet» із обов'язковою відеофіксацією процесу проведення іспиту.

## СПИСОК РЕКОМЕНДОВАНОЇ ЛІТЕРАТУРИ

1. Кублій, Л. І. Алгоритми та структури даних. Основи алгоритмізації [Електронний ресурс] : підручник для здобувачів ступеня бакалавра за спеціальністю 121 «Інженерія програмного забезпечення» / Л. І. Кублій ; КПІ ім. Ігоря Сікорського. – Електронні текстові дані (1 файл: 21,3 Мбайт). – Київ : КПІ ім. Ігоря Сікорського, 2022. – 528 с. – Назва з екрана.

2. Ірина Бородкіна Теорія алгоритмів. Посібник для студентів вищих навчальних закладів. – К.: Центр навчальної літератури, 2019. – 184 с.

3. Cormen, T. H., Leiserson, C. E., Rivest, R. L., та ін. Introduction to Algorithms: Cambridge, MA: The MIT Press, 2022. 1312 с.

4. Sedgewick, Algorithms In C: Fundamentals, Data Structures, Sorting, Searching, Parts 1-4, 3/E – Pearson Education, 1998. – 720p.

5. Олексій Васильєв. Програмування мовою Java. – К.: Навчальна книга – Богдан, 2020. – 696 с.

6. Fundamentals of Programming [Електронний ресурс]. – Режим доступу: <http://route.ho.ua/kpi/c1/>

7. Fundamentals of Programming-II [Електронний ресурс] – Режим доступу: <http://route.ho.ua/kpi/c2/>

8. Herbert Schildt Java: The Complete Reference, Twelfth Edition 12th Edition – McGraw Hill, 2021. – 1280 p.

9. Bjarne Stroustrup Tour of C++, A (C++ In-Depth Series) 2nd Edition – Addison-Wesley Professional, 2018. – 256p.

10. Gamma E. Design Patterns: Elements of Reusable Object-Oriented Software —Addison-Wesley Professional, 1994. – 416p.

11. Grady Booch Object-oriented analysis and design with applications —Addison Wesley, 2007. – 720 p.

12. Craig Larman Applying UML and Patterns: An Introduction to Object-oriented Analysis and Design and Iterative Development – Prentice Hall PTR, 2005. – 736p.

13. David A. Marca Sadt: Structured Analysis and Design Techniques – McGraw-Hill, 1987. – 392 p.

**ПЕРЕЛІК РОЗРОБНИКІВ:**

д.т.н., проф. каф. ПЗЕ



Олексій НЕДАШКІВСЬКИЙ

к.т.н., доц. каф. ПЗЕ



Іван ВАРАВА

д.філософії., ст.викл. каф. ПЗЕ



Вадим КОЛУМБЕТ

Програму рекомендовано:

кафедрою інженерії програмного забезпечення в енергетиці НН ІАТЕ

Протокол № 26 від 21 лютого 2025 р.

В.о. зав. кафедрою інженерії  
програмного забезпечення  
в енергетиці



Олександр КОВАЛЬ