



## **Зміст**

1. Загальні відомості
2. Питання, що виносяться на вступне випробування
3. Навчально-методичні матеріали
4. Рейтингова система оцінювання
5. Приклад екзаменаційного білету

## 1. Загальні відомості

Ця Програма призначена для вступників на навчання для здобуття ступеня доктора філософії за спеціальністю 121 Інженерія програмного забезпечення інформації (освітня програма «Інженерія програмного забезпечення»).

Метою Програми є надання вступникам інформації щодо змісту навчального матеріалу, перевірка знань з якого відбувається під час фахового вступного випробування для вступу на навчання для здобуття ступеня доктора філософії за спеціальністю 121 Інженерія програмного забезпечення (освітня програма «Інженерія програмного забезпечення»).

Фахове вступне випробування для вступу на навчання для здобуття ступеня магістра за спеціальністю 121 Інженерія програмного забезпечення (освітня програма «Інженерія програмного забезпечення») складається з двох частин: письмової відповіді на три питання та співбесіди для уточнення змісту письмової відповіді. Екзаменаційний білет містить 3 питання.

Зміст навчального матеріалу, перевірка знань з якого відбувається під час фахового вступного випробування, наведений у розділі 2 «Питання, що виносяться на вступне випробування» цієї Програми.

Для підготовки до випробування рекомендовано використовувати навчальну літературу, наведену у розділі 3 «Навчально-методичні матеріали».

Тривалість випробування – 90 хвилин (для відповіді на кожне з 3 питань надається по 30 хвилин). Інформація щодо методики оцінювання наведена у розділі 4 «Рейтингова система оцінювання». Приклад білету наведено у розділі 5 «Приклад екзаменаційного білету».

## 2. Питання, що виносяться на вступне випробування

### 1. Математичні основи інженерії програмного забезпечення

1. Математичні основи криптографії. Типи незворотних перетворень, їх порівняльна характеристика та використання в алгоритмах захисту інформації. Використання булевих перетворень для задач криптографічного захисту інформації. Криптографічні властивості булевих перетворень.
2. Алгоритми потокового шифрування даних. Типи генераторів псевдовипадкових двійкових послідовностей. Методи оцінки їх якості. Методи злomu захисту з використанням потокового шифрування.
3. Криптографічні алгоритми симетричного шифрування. Структури алгоритмів симетричного шифрування. Диференційний та лінійний криптоаналіз.
4. Хеш-алгоритми і їх використання для криптографічного захисту цілісності та автентичності електронних документів. Структури хеш-алгоритмів та методи їх злomu.
5. Механізми криптографічного захисту на основі теорії чисел. Алгоритм RSA. Генерація ключів алгоритму. Методи злomu криптографічного захисту з використанням аналізу динаміки параметрів технічної реалізації.
6. Методи організації обчислення модулярного експоненціювання при реалізації криптографічних алгоритмів з відкритим ключем. Технологія Монтгомері та використання передобчислень. Обчислення модулярної експоненти з захистом від аналізу динаміки споживання потужності.
7. Цифровий підпис. Алгоритми його формування та перевірки. Методи підробки цифрового підпису.
8. Методи та криптографічні механізми ідентифікації віддалених користувачів. Атаки на системи ідентифікації. Концепція нульових знань при ідентифікації віддалених користувачів та криптографічні алгоритми її реалізації.
9. Інтерполяція і наближення функцій. Інтерполяційний поліном Лагранжа.
10. Безперервні випадкові величини. Функція щільності ймовірності та її властивості. Ймовірність влучення на заданий відрізок.
11. Дискретні випадкові величини. Ряд розподілу дискретної СВ. Функція розподілу, її властивості і графік. Ймовірність попадання на напівінтервал.

12. Нормальний розподіл. Крива Гауса. Функція розподілу нормального закону. Правило «трьох сігм».
13. Функція декількох випадкових величин. Розподіл суми і добутку незалежних випадкових величин. Числові характеристики випадкових величин: математичне сподівання, дисперсія.
14. Загальні положення та основні поняття дослідження операцій. Класифікація екстремальних задач. Основні етапи розв'язання екстремальних задач
15. Транспортні задачі лінійного програмування. Постановка, основні властивості транспортної задачі та методи знаходження початкового базисного розв'язку. Метод потенціалів та угорський метод.
16. Нелінійне програмування. Двоїстість в задачах оптимізації та методи знаходження умовних екстремумів. Квадратичне програмування. Геометричне та дробово-лінійне програмування.
17. Чисельні методи нелінійної оптимізації. Градієнтні методи та методи змінної метрики. Метод проєкції градієнта Розена, метод зведеного градієнта та метод штрафних функцій. Мінімізація негладких функцій.
18. Динамічне програмування. Основна ідея та особливості обчислювального методу динамічного програмування. Динамічне програмування для задач з декількома обмеженнями та змінними. Динамічні задачі управління запасами та динамічне програмування для марковських процесів.
19. Системи масового обслуговування. Основні поняття теорії масового обслуговування. Моделі народження і загибелі та загальна модель системи масового обслуговування
20. Методи оптимізації в задачах великої вимірності. Методи декомпозиції. Декомпозиція Корнаї-Ліптака.
21. Елементи теорії ігор. Основні поняття та визначення в області теорії ігор. Позиційні ігри. Теорема про мінімакс Дж. фон Неймана. Застосування лінійного програмування для розв'язків матричних ігор.

## ***2. Алгоритмічні основи інженерії програмного забезпечення***

1. Частково рекурсивні функції. Рекурсивні предикати. Примітивно рекурсивні функції.
2. Марковські підстановки. Нормальні алгоритми Маркова. Нормально обчислювані функції і принцип підстановки Маркова.
3. Постановка завдання кластеризації. Функції відстані. Критерії якості кластеризації. Алгоритм К-середніх та модифікації.

4. Алгоритми пошуку найкоротшого шляху. Задача про найкоротші шляхи з одного входу. Пошук дерева найкоротшого шляху. Алгоритм Дійкстри.
5. Задача про найкоротші шляхи для всіх пар. Динамічне програмування для знаходження найкоротшого шляху. Алгоритм Флойда-Воршелла. Алгоритм Джонсона.
6. Формалізація постановки задачі в просторі станів. Стратегії сліпого пошуку. Ітерацій не поглиблення. Особливості, переваги і недоліки цих стратегій.
7. Приклади семантичних мереж. Концепція. Основні поняття, терміни, переваги і недоліки фреймової моделі. Приклад фреймової моделі.
8. Логічне програмування. Факти, правила та питання. Теорія логічного програмування.
9. Інтуїтивне визначення алгоритму, вирішуваність, обчислюваність, перераховність. Способи подання алгоритмів.
10. Сутність та особливості оцінювання складності алгоритмів. Теоретичні основи. Види оцінок складності.
11. Формування знань на основі машинного навчання. Індуктивне навчання. Дерева рішень.
12. Динамічне програмування. Загальна схема. Умови застосування. Реалізація схеми динамічного програмування для задач множення матриць
13. Жадібні алгоритми. Загальна схема. Умови застосування. Теоретичні основи жадібних алгоритмів. Жадібні алгоритми для задач про вибір заявок і розклад, побудови кодів Хафмена,
14. Алгоритми паралельних обчислень. Моделі паралельних машин. Ефективна паралельна обробка префіксів.
15. Мережі і проблема визначення кістякового дерева мінімальної ваги. Алгоритм PRIM. Реалізація і аналіз складності алгоритму.
16. Алгоритми сортування і пошуку даних. Класифікація. Переваги і недоліки. Класичні («бульбашка», вставками, Шела тощо) і сучасні алгоритми сортування.
17. Алгоритми пошуку підрядків. Алгоритми Рабіна-Карпа, Кнута-Моріса-Прата, Боєра-Мура. Алгоритми пошуку підрядків на основі скінчених автоматів.
18. Базові структури даних. Масиви. Зв'язані списки. Списки суміжності. Стеки, черги. Двійкові дерева пошуку. Випадкові двійкові дерева пошуку.
19. Розвиток уявлень про структури даних. Загальна характеристика хешування. Хеш-функції. Відкрита адресація. Організація переповнення

- ланцюжками. Червоно-чорні дерева: визначення; операції; оцінки; застосування. Б-дерева: визначення; операції; оцінки; застосування.
20. Уточнення інтуїтивного визначення алгоритму за Т'юрингом. Розв'язувані і не розв'язувані предикати, обчислювальні і не обчислювальні функції. Теорема Черча.
  21. Рекурсивні функції. Визначення і властивості примітивно-рекурсивних, частково-рекурсивних і загально-рекурсивних функцій.
  22. Рекурсивна обчислюваність. Рекурсивна вирішувальність, примітивно-рекурсивні множини, рекурсивні множини, частково-рекурсивні множини. Властивості рекурсивних і примітивно-рекурсивних множин. Рекурсивно-перераховні множини.
  23. Теорія складності. Проблеми і кодування, обчислювальна модель класи P, NP. Базові поняття теорії звідності, співвідношення класів P, NP і NP-повних проблем, звідність, теорема Кука. Елементи теорії NP-повноти. Проблеми 3-виконуваність, «тривимірне сполучення», «вершинне покриття», «Кліка», розбиття, «Гамільтонів цикл», ієрархія NP-повних проблем. Методи доведення NP-повноти. Загальна характеристика прийомів доведення NP-повноти, звуження проблеми, локальна заміна, побудова компонентів, Аналіз підпроблем. Проблеми з числовими параметрами і сильна NP-повнота.
  24. Скінчені автомати, їх аналіз і синтез. Автомати з магазинною пам'яттю. Машини Т'юринга і лінійно-обмежені автомати. Автомати і граматики.

### ***3. Архітектура програмного забезпечення***

1. Основні поняття програмної інженерії. Основні етапи життєвого циклу програмних систем. Варіанти (моделі) життєвого циклу. Загальна характеристика етапів життєвого циклу програмних систем. Характеристика моделей життєвого циклу програмного продукту: каскадна, спіральна, ітеративна, інкрементна (пошагова), формальна, швидкого прототипування. Їх переваги, недоліки та використання.
2. Визначення та задачі аналізу вимог до програмного забезпечення. Вимоги до продукту та процесу. Функціональні та нефункціональні вимоги. Вимоги з кількісною оцінкою. Системні та програмні вимоги. Моделі процесу визначення вимог до програмного забезпечення. Типи учасників процесу та їх роль. Джерела вимог та техніки отримання вимог.
3. Структура та архітектура програмного забезпечення. Архітектурні точки зору та представлення. Характеристика архітектур програмних продуктів. Поняття шаблонів проектування, фреймворки. Загальні концепції проектування програмного забезпечення. Концепції та процес

- проектування програмного забезпечення. Загальні принципи (техніки) проектування.
4. Фундаментальні питання та проблеми проектування програмного забезпечення. Паралелізм та ефективна обробка програмних процесів. Контроль та обробка подій. Розподіл компонентів. Обробка помилок та виключних ситуацій. Взаємодія системи з користувачами та збереження даних.
  5. Моделювання у програмній інженерії. Основні принципи моделювання програмних систем. Візуальне моделювання та CASE-технології. Уніфікована мова моделювання *UML*.
  6. Призначення *UML*. Види діаграм *UML*. Діаграми структури та діаграми поведінки. Спрощена стратегія використання *UML*-діаграм при моделюванні програмного забезпечення.
  7. Діаграми прецедентів та їх використання. Моделювання контексту та вимог до програмної системи. Прецеденти та потоки подій. Описи потоків подій. Потоки подій та сценарії як типи й екземпляри. Специфікація прецедентів. Використання сценаріїв при плануванні версій. Зв'язки між акторами та прецедентами. Організація прецедентів (відношення залежності, включення та розширення). Відношення узагальнення для прецедентів та акторів.
  8. Реалізація прецедентів та використання діаграм послідовностей. Побудова діаграм послідовності. Двоетапне розроблення діаграм послідовностей. Діаграми співробітництва.
  9. Узгодженість (цілісність) моделей. Виявлення класів. Класи етапу аналізу: прикордонні (boundary) або інтерфейсні класи; класи-сутності (entity); управляючі (control) класи (класи-менеджери). Класи етапу проектування.
  10. Відношення між класами (узагальнення, залежність, асоціація, агрегація, композиція) та їх виявлення. Проектування класів, відношень між класами. Пакетування класів.
  11. Класифікація патернів: породжуючі патерни (патерни процесу створення об'єктів); структурні патерни (що ґрунтуються на композиціях – структурних об'єднаннях об'єктів чи класів); патерни поведінки (що характеризуються взаємодією об'єктів між собою).
  12. Успадкування та композиції. Простір патернів проектування. Приклади патернів. Кодогенерація (інженірінг) та реінженірінг.
  13. Стратегії та методи проектування програмного забезпечення. Загальні стратегії. Функціонально-орієнтоване та структурне проектування. Об'єктно-орієнтоване проектування. Проектування на основі структур даних. Компонентне проектування.



- 14.Будова програми при використанні процедурного (імперативного) програмування. Проблеми, що виникають при використанні процедурного програмування та їх вирішення використанням об'єктно-орієнтованого програмування.
- 15.Характеристика структурного програмування як методології розробки програмного забезпечення. Базові конструкції програми. Поняття підпрограми (процедури або функції). Метод розробки програми. Мета та принципи модульного програмування.
- 16.Методології розроблення програмного забезпечення Microsoft Solution Framework, Rational Unified Process, Extreme Programming. Засоби розробки, що базуються на них.
- 17.Мета та задачі супроводу програмного забезпечення. Еволюція програмного забезпечення та потреба у його супроводі. Категорії та ключові питання супроводу програмного забезпечення.
- 18.Мета та задачі управління програмними проектами. Аналіз здійсненності програмного проекту: технічні, операційні, фінансові та соціальні аспекти. Процес оцінки та перегляду вимог до програмного забезпечення. Планування програмного проекту. Управління виконанням програмного проекту.
- 19.Поняття якості програмного забезпечення. Характеристика основних критеріїв якості програмного продукту. Основні принципи та задачі керування якістю програмного забезпечення. Характеристика дефектів та методи управління якістю програмного забезпечення.
- 20.Аналіз якості та оцінювання програмного проектування. Показники якості розробки програмного забезпечення. Аналіз якості та техніки оцінювання. Виміри в проектуванні програмного забезпечення.
- 21.Основні принципи конструювання програмного забезпечення: мінімізація складності, очікування змін, конструювання з можливістю перевірки, стандарти конструювання.
- 22.Загальні принципи керування конструюванням програмного забезпечення. Планування та вимірювання в конструюванні програмного забезпечення. Мови конструювання та програмне кодування. Тестування та якість конструювання. Загальні аспекти повторного використання.
- 23.Інструменти програмної інженерії: інструменти для роботи з вимогами, інструменти для проектування, інструменти конструювання, інструменти тестування, інструменти супроводу, інструменти конфігураційного керування та керування інженерною діяльністю.
- 24.Методи програмної інженерії. Евристичні методи. Формальні методи. Методи прототипування.

25. Технологія HTML5 Web Components причини її виникнення. Постановка завдань і способи їх вирішення за допомогою технології HTML5 Web Components.
26. Проблема конфліктів стилів під час розробки інтерфейсу користувача у мережевому програмуванні. Використання Shadow DOM для вирішення цієї проблеми.
27. Селектори CSS для спрощення оформлення графічних елементів. Способи об'єднання JavaScript, HTML и CSS коду у один документ.
28. Сучасні технології для розробки веб-додатків, що побудовані на технології (патерну) MVC. Привести приклад актуальної технології, показати на прикладі архітектури, мови що використовуються. Реалізація маршрутизації, зовнішнього вигляду, моделі і контролера.
29. Тестування програмного забезпечення. Фундаментальні положення про тестування ПЗ. Термінологія та ключові питання тестування програмного забезпечення. Зв'язок тестування з іншими видами діяльності процесу розроблення програмного забезпечення. Мета та рівні тестування.
30. Техніки тестування та їх комбінація. Вимір результатів тестування. Рівні тестування (SWEBOOK. Guide V3.0). Тестування програмного забезпечення. Техніки тестування ПЗ. Тестування програмного забезпечення. Оцінка рівня тестування ПЗ (SWEBOOK. Guide V3.0). Інструментальні засоби для тестування програмного забезпечення. Процес тестування (SWEBOOK. Guide V3.0).

#### ***4. Мови та технології програмування***

1. Структура програм на мовах програмування Python, C/C++ або Java. Прості та складені оператори, приклади використання операторів. Поняття константи та змінної в мовах програмування, принципи виділення пам'яті для розміщення констант та змінних. Оголошення та ініціалізація констант і змінних у програмах на мовах програмування Python, C/C++ або Java. Прості типи даних. Розміщення даних різних типів в оперативній пам'яті. Приклади оголошення різних типів даних у програмах на мовах програмування Python, C/C++ або Java. Сумісність типів та приведення типів даних. Типи даних користувача. Оголошення та використання таких типів в мовах програмування Python, C/C++ або Java. Операції, вирази та оператори в мовах програмування Python, C/C++ або Java. Операції привласнення та арифметичні операції. Пріоритети операцій. Зміна пріоритетів. Поняття глобальних та локальних змінних та приклади їх використання в програмах на мовах Python, C/C++ або Java. Статичні та динамічні змінні.

2. Логічні операції та умовні оператори. Алгоритмічна структура розгалуження. Вибір із двох альтернатив. Блок схеми структур розгалуження. Вкладеність конструкцій вибору.
3. Алгоритмічні конструкції повторення. Типи циклів. Блок схеми циклів. Організація циклів.
4. Поняття функції (підпрограми) в мовах програмування. Оголошення та виклик функцій. Рекурсія функцій. Стандартні процедури та функції.
5. Поняття масиву та його властивості, їх оголошення і використання. Одновимірні та багатовимірні масиви. Доступ до елементів масиву.
6. Рядки. Операції над символами та рядками в мовах програмування Python, C/C++ або Java. Стандартні процедури та функції обробки рядків.
7. Записи та структури, їх оголошення і використання. Доступ до компонентів та операції над записами та структурами. Похідні типи даних.
8. Основні концепції об'єктно-орієнтованої методології програмування. Програмна модель об'єкту. Наведіть приклад програмної моделі довільного об'єкту на мові програмування Python, C++ або Java.
9. Оголошення класів на мові програмування Python, C++ або Java. Конструктори і деструктори класу. Призначення і основні правила використання. Наведіть приклад оголошення довільного класу та його використання в тексті програм на мові програмування.
10. Структура програми, що використовує класи, на мові програмування Python, C++ або Java. Наведіть приклади.
11. Особливості мови програмування Python, синтаксис, напрями використання, розповсюдженість. Привести приклади коду з коментарями.
12. Особливості мови програмування PHP, синтаксис, напрями використання, розповсюдженість. Привести приклади коду з коментарями.
13. Особливості мови програмування Java, синтаксис, напрями використання, розповсюдженість. Привести приклади коду з коментарями.
14. Поняття інкапсуляції та її застосування в мовах програмування. Методи які організують інтерфейси. Наведіть приклади визначення відкритих закритих полів і методів класу.
15. Поняття спадкування та поліморфізму в об'єктно-орієнтованих мовах програмування. Поняття та приклади перевантажених функцій. Ієрархія класів. Одиночне і множинне спадкоємство.

16. Абстрактні класи. Призначення і використання. Наведіть приклад абстрактного класу. Спеціальні методи класів. Перевантаження операторів. Поняття дружніх функцій і класів. Особливість доступу до закритих полів класу. Наведіть приклади організації доступу до закритих полів.
17. Організація обмінних операцій з використанням бібліотеки класів для введення - виведення. Поняття файлу і потоку, приклади роботи з ними. Маніпулятори потоків.
18. Оброблення програмних помилок засобами мови програмування. Алгоритмічна конструкція типу try catch. Приклади використання оброблення помилок роботи програмного блоку.
19. Поняття шаблону функції та класу. Абстрактні алгоритми. Бібліотека стандартних шаблонів STL (Standard Template Library).
20. Компілятори і інтерпретатори. Опис їх роботи та порівняння переваг і недоліків. Основні задачі, що вирішуються компілятором. Вхідні і вихідні файли, їх логічний зміст.
21. Поняття про ресурси комп'ютера. Операційна система як програма управління ресурсами. Основні завдання управління програмними процесами, оперативною пам'яттю і пристроями введення - виводу в операційній системі.
22. Поняття багатозадачної операційної системи. Витисняюча багатозадачність. Схема витисняючої багатозадачності квантуванням за часом. Управління процесами в багатозадачних ОС. Основні алгоритми управління. Управління оперативною пам'яттю в багатозадачній ОС. Основні алгоритми управління. Віртуальна пам'ять. Управління ресурсами. Монопольне та спільне використання ресурсів. Поняття моделювання ресурсу. «Спулінг». Критичні секції в асинхронних процесах багатозадачних ОС. Основні алгоритми синхронізації. Завдання синхронізації процесів. Поняття критичного шляху. «Змагання» і «клінчі».
23. Основні методи організації зберігання файлів на зовнішніх пристроях прямого методу доступу. Поняття бібліотечної, індексної послідовної і кластерної структури.
24. Типи діалогів, порівняльні характеристики діалогів. Групи користувачів ОС. Основні характеристики діалогів типу меню та команда. Групи користувачів ОС.
25. Особливості мови програмування C#, синтаксис, напрями використання, розповсюдженість. Привести приклади коду з коментарями.
26. Особливості мови програмування C++, синтаксис, напрями використання, розповсюдженість. Привести приклади коду з коментарями.

27. Особливості мови програмування C, синтаксис, напрями використання, розповсюдженість. Привести приклади коду з коментарями.
28. Особливості мови програмування JavaScript, синтаксис, напрями використання, розповсюдженість. Привести приклади коду з коментарями.
29. Способи досягнення надійності програмного забезпечення. Привести практичні приклади з використанням базової мови програмування. Поняття про тестування. Навести простий приклад тестування.
30. Методологія розробки програмного забезпечення. Технологічний процес створення програмного забезпечення. Моделі життєвого циклу розробки програмного забезпечення. Ступенчаста і водоспадна модель. Пояснити на прикладі. Переваги і недоліки моделі. Спіральна модель. Пояснити на прикладі. Переваги і недоліки моделі. Раціональний уніфікований процес. Пояснити на прикладі. Переваги і недоліки моделі. Екстремальне програмування.
31. Основи .NET. Структура платформи .Net., версії .Net. Структура .Net Framework, версії .Net Framework. Віртуальна машина CLR. Структура та принцип роботи. Загальна система типів CTS. Перетворення типів. Управління ресурсами пам'яті в .Net. WPF та XAML.
32. Основні синтаксичні конструкції C#. Робота з рядками. Робота з масивами даних. Робота з файлами та директоріями. Серіалізація даних.
33. Способи реалізації принципів ООП у мові C#. Сутність об'єктно-орієнтованого підходу до створення ПЗ. Класи. Інкапсуляція. Наслідування. Поліморфізм. Інтерфейси. Абстрактні класи. Колекції. Делегати. Події. Обробка виключних ситуацій та їх генерація користувачем.
34. Шаблони проектування. Структурні шаблони. Шаблон «Декоратор». Шаблон «Компонувальник». Шаблон «Міст». Шаблон «Адаптер». Шаблон «Пристосуванець». Шаблони «Фасад», «Заступник». Породжувальні шаблони. Шаблон «Прототип». Шаблон «Одинак». Шаблон «Фабричний метод». Шаблон «Абстрактна фабрика». Шаблон «Будівельник». Поведінкові шаблони. Шаблон «Стратегія». Шаблон «Шаблонний метод». Шаблон «Спостерігач». Шаблон «Ланцюжок обов'язків». Шаблон «Команда». Шаблон «Посередник». Шаблон «Ітератор».

## ***5. Засоби створення інформаційних систем***

1. Нормалізація баз даних. Перша і друга нормальні форми. Поняття нормалізації бази даних. Використання методу нормальних форм для оптимізації структури відношень. Залежності між атрибутами.

Виявлення залежностей між атрибутами. Критерій першої нормальної форми, поняття та приклади повторюваних груп та способу їх усунення. Критерій другої нормальної форми, поняття та приклади функціональної залежності атрибутів та визначника (детермінанта). Критерій третьої нормальної форми, поняття та приклади транзитивної залежності між атрибутами відношення. Критерій нормальної форми Бойса-Кодда.

2. Визначення поняття інтерфейс користувача, типи користувацьких інтерфейсів. Принципи якості інтерфейсів. Тенденції розвитку сучасних веб-інтерфейсів. Каркаси веб-застосунків, засоби їх створення. Каркасне моделювання інтерфейсів веб-застосунків. Мови та платформи створення веб-застосунків та інтерфейсів.
3. Поняття «хмарні обчислення». Характеристики хмарних обчислень. Можливості хмарних обчислень. Переваги та недоліки. Класифікація хмарних сервісів. Моделі розгортання інфраструктури хмарних обчислень, визначення кожної з них. Моделі обслуговування, визначення, сучасні рішення.
4. Конкурентні системи. Паралелізм. Моделі конкурентного програмування. Sharedmemory. Модель обміну повідомленнями. Проблеми гонки за ресурси (racescondition).
5. Розподілені системи. Синхронізація часу. Алгоритм Крістіана. Алгоритм Берклі. Транзакція. Принципи ACID. Архітектура MapReduce.
6. Інтелектуальні методи обчислювання для кластеризації, класифікації, моделювання та прогнозування.
7. Сучасні СУБД та інтегровані служби аналізу даних, можливості інтеграції служб у корпоративні інформаційні системи. Поняття бази даних та системи управління базами даних (СУБД). Поняття сутності, атрибуту, зв'язку та домену та їх ілюстрація на довільному прикладі. Найбільш поширені СУБД.
8. Проблема надлишкового дублювання та узгодження введення дубльованих даних. Вимоги до вибору ідентифікаторів записів. Аномалії вставки, видалення та модифікації даних. Поняття цілісності даних.
9. Базові типи зв'язків між даними. Формування зв'язків типів "один до одного", "один до багатьох" та "багато до багатьох". Приклади даних із такими типами зв'язків.
10. Елементи моделі сутність-зв'язок. Основні визначення ER-діаграм. Приклади ER-діаграм з різноманітними типами зв'язків.
11. Мова запитів до баз даних SQL. Особливості застосування. Альтернативні мови запитів до баз даних. Особливості застосування. Команди маніпуляції із таблицями: команда додавання стовпців,

команда модифікації характеристик стовпців, команда перейменування та команда видалення стовпців, команда видалення таблиць. Синтаксис оператора SELECT. Приклади виразу SQL, що організує вибірку даних із двох таблиць для певного критерію. Синтаксис операторів INSERT, DELETE, UPDATE. Приклади виразів SQL, що організують вставку нових даних, видалення та модифікацію даних, відповідно. Використання вкладених підзапитів в мові SQL. Принцип роботи запиту зі зв'язаним підзапитом. Приклад організації запиту з підзапитом. Використання HAVING в зв'язаних підзапитах. Оператор об'єднання UNION. Принцип роботи запиту при використанні оператору. Приклад використання оператора UNION. Усунення дублювання в UNION. Операції поєднання таблиць шляхом посилальної цілісності. Різновиди та принципи роботи оператору об'єднання таблиць JOIN. Приклади використання оператора JOIN. Використання індексування та розділення для оптимізації продуктивності баз даних. Приклад атрибутів відношень, для яких доцільна індексація та приклад виразу SQL для створення індексу.

12. Організація даних та загальні відомості про моделі даних. Порівняльна характеристика реляційної та об'єктно-орієнтованої моделей даних. Приклади СУБД, що використовують такі моделі. Мережеві та ієрархічні моделі даних. Реляційна модель даних. Система баз даних. Забезпечення незалежності даних. Архітектура системи баз даних та її рівні. Архітектура "клієнт-сервер". Суть розподіленої обробки даних в комп'ютерних системах та мережах.
13. Системи управління базами даних (СУБД). Основні ознаки класифікації СУБД. Основні засоби СУБД. Властивості СУБД та технології їх використання. Об'єктно-орієнтовані СУБД. Об'єкти та об'єктні класи. Адміністрування даних та адміністрування баз даних. Основні етапи розробки бази даних. Інформаційно-логічна модель даних предметної області та технологія її розробки. Визначення логічної структури бази даних.
14. Класи сучасних процесорів. Загальні та відмінні риси мікроконтролерів та процесорів загального застосування. Поняття убудованої системи. Приклади архітектур сучасних процесорів.
15. Дати визначення технології Інтернету речей (Internet of Things, IoT). Стек технологій, які використовуються в рамках IoT, навести приклади сервісів IoT.
16. Загальні принципи передачі файлів та повідомлень через комп'ютерні мережі, переваги пакетної передачі. Характеристика рівням моделі взаємодії відкритих систем (Open System Interconnect - OSI), приклади протоколів різних рівнів.

17. Формат MAC-адреси, поняття однопунктових (Unicast), багатопунктових (Multicast) та ширококомовних (Broadcast) MAC-адреси. Приклади використання протоколами кадрів із такими типами MAC-адрес.
18. Поняття колізії при організації колективного доступу до середовища передачі в технології Ethernet. Алгоритм метода колективного доступу до середовища передачі з розпізнанням несучої та виявленням колізій (Carrier-Sense-Multiply-AccesswithCollisionDetection — CSMA/CD).
19. Загальна характеристика технології Ethernet. Успадковані технології Fast Ethernet, Gigabit Ethernet і 10 Gigabit Ethernet, їх спільні риси.
20. Загальна характеристика протоколів мережного рівня. Формат IP-адреси та принцип виділення блоків IP-адрес за класовою схемою. Особливі IP-адреси та правила їх вживання. Планування IP-підмереж за допомогою масок підмереж.
21. Відображення IP-адрес на адреси канального рівня за допомогою протоколу ARP. Модель роботи ARP. Відображення доменних імен на IP-адреси, етапи процесу визначення IP-адреси за доменним ім'ям.
22. Основні задачі протоколу IP. Формат заголовку та характеристика полів, що вирішують задачі IP.
23. Принципи маршрутизації IP-пакетів у складеній мережі, структура таблиць маршрутизації та команди задавання статичного маршруту до мереж.
24. Протоколи динамічної маршрутизації. Поняття, визначення, основні алгоритми роботи.
25. Загальна характеристика та основні задачі протоколу контрольних повідомлень Інтернету ICMP. Типи повідомлень ICMP. Принципи використання ICMP повідомлень в утилітах ping та traceroute.
26. Задачі протоколів транспортного рівня. Порівняльна характеристика протоколів TCP та UDP. Поняття портів та адрес сокетів, розподіл номерів портів. Номери портів популярних протоколів прикладного рівня.
27. Процедури встановлення з'єднання, передачі даних та завершення з'єднання протоколом TCP. Поля та прапорці заголовків пакетів, що використовуються при цих процедурах.
28. Формат заголовку TCP. Використання полів та прапорців для цілісної та гарантованої передачі даних. Спосіб керування потоком даних за допомогою "вікна".
29. Поняття телекомунікацій. Структура територіальної мережі. Приклади протоколів та технологій територіальних мереж, побудованих на комутації віртуальних каналів та комутації пакетів/комірок.



30. Загрози інформаційній безпеці комп'ютерної обробки даних. Основні задачі захисту інформації в комп'ютерних системах та мережах. Організаційні, технічні та криптографічні засоби захисту інформації. Засоби обмеження доступу до інформації, використання брандмауерів для захисту комп'ютерних мереж від спроб несанкціонованого доступу.
31. Комп'ютерні віруси, їх класифікація, способи розповсюдження вірусів. Методи виявлення комп'ютерних вірусів та захисту від них.
32. Штучний інтелект, поняття, принципи, класи задач, алгоритми. Машинне навчання, поняття, принципи, класи задач, алгоритми. Нейронні мережі, моделі. Навчання з вчителем та без. Глибоке навчання (Deep Learning).
33. Хмарні обчислення (принципи побудови, особливості, переваги, недоліки, приклади, перспективи розвитку). Моделі розгортання в контексті хмарних обчислень. Моделі обслуговування в контексті хмарних обчислень. Віртуалізація в контексті хмарних обчислень. Розподілена файлова система в контексті хмарних обчислень. Тенденції розвитку розподілених систем. Приклади хмарних сервісів Microsoft, Google, Amazon. Квантові комп'ютери.

### 3. Навчально-методичні матеріали

1. Настенко, Д. В. Об'єктно-орієнтоване програмування. Частина 1. Основи об'єктно-орієнтованого програмування на мові С# / Д. В. Настенко, А. Б. Нестерко. – Київ : НТУУ «КПІ», 2016. – 76 с. – Режим доступу: <http://ela.kpi.ua/handle/123456789/1667>
2. Бублик В.В. Об'єктно-орієнтоване програмування: [Підручник] / В.В. Бублик. – К.: ІТ-книга, 2015. – 624 с.: іл. – ISBN 978-966-97182-1-1. – Режим доступу: [http://itknyga.com.ua/docs/OOP\\_final.pdf](http://itknyga.com.ua/docs/OOP_final.pdf)
3. Жуковський С.С., Вакалюк Т.А. Об'єктно-орієнтоване програмування мовою С++. – Житомир : Вид-во ЖДУ, 2016. – 100с. – Режим доступу: [http://lib.iitta.gov.ua/706370/1/%D0%BF%D0%BE%D1%81%D1%96%D0%B1%D0%BD%D0%B8%D0%BA\\_%D0%A1\\_\\_OOP1.pdf](http://lib.iitta.gov.ua/706370/1/%D0%BF%D0%BE%D1%81%D1%96%D0%B1%D0%BD%D0%B8%D0%BA_%D0%A1__OOP1.pdf)
4. Ю.П. Зайченко. Дослідження операцій. Підручник. Сьоме видання, перероблене та доповнене. – К.: Видавничий Дім “Слово”, 2006. – 816 с.
5. Зайченко О.Ю., Зайченко Ю.П. Дослідження операцій. Збірник задач. – К.: Видавничий дім “Слово”, 2007. – 472 с.
6. Кутковецький В.Я. Дослідження операцій: Навчальний посібник. – Київ: Вид-во ТОВ “Видавничий дім “Професіонал”, 2004. – 350 с.
7. Катренко А.В. Дослідження операцій: підручник. – Львів: МагноліяПлюс, 2004. – 549 с.
8. Горбань О.М. Основи теорії систем та системного аналізу [Текст]: / Горбань О.М., Бахрушин В.Є. – Запоріжжя: ГУ "ЗІДМУ", 2004.

#### 4. Рейтингова система оцінювання

При вступі на третій (освітньо-науковий) рівень вищої освіти для здобуття наукового ступеня доктор філософії вступний іспит складається з двох частин: письмової відповіді на три питання та співбесіди для уточнення змісту письмової відповіді. Екзаменаційний білет містить три питання. Відповідь на перше та друге питання білета оцінюється за 30-бальною системою (табл. 1), а на третє питання оцінюється за 40-бальною шкалою (табл. 2). Оцінка, яку вступник отримує за відповідь на кожне питання, визначається за системою балів, поданою нижче. Критерії оцінювання наведені в таблицях 1 та 2.

Таблиця 1

Бали	Характеристика відповіді
30	Повна правильна відповідь на запитання з відповідними поясненнями.
25-29	Відповідь правильна, але не зовсім повна.
19-24	Відповідь в цілому правильна, але містить окремі неточності.
12-18	Відповідь в цілому правильна, але містить неточності або відповідь неповна.
6-11	Відповідь в цілому правильна, але містить суттєві неточності або відповідь суттєво неповна.
1-5	Відповідь містить суттєві помилки.
0	Відповідь на питання відсутня.

Таблиця 2

Бали	Характеристика відповіді
40	Повна правильна відповідь на запитання з відповідними поясненнями.
35-39	Відповідь правильна, але не зовсім повна.
28-34	Відповідь правильна, але не повна (відсутня одна позиція відповіді).
21-27	Відповідь в цілому правильна, але містить окремі неточності.
13-20	Відповідь в цілому правильна, але містить неточності або відповідь неповна.
6-12	Відповідь в цілому правильна, але містить суттєві неточності або відповідь суттєво неповна.
1-5	Відповідь містить суттєві помилки.
0	Відповідь на питання відсутня.

Початковий рейтинг абітурієнта за екзамен розраховується виходячи із 100-бальної шкали:  $30+30+40=100$  балів. При визначенні загального рейтингу вступника початковий рейтинг за екзамен перераховується у 200-бальну шкалу за табл. 3.

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

Вступне випробування проводять лише за затвердженим комплектом екзаменаційних білетів. Відмова студента від написання вступного випробування за екзаменаційним білетом атестується як незадовільна відповідь.

Під час вступного випробування дозволяється користуватися ручкою та листами вступного випробування. При виявленні факту використання недозволених матеріалів екзаменаційна комісія має право припинити випробування і виставити незадовільну оцінку.

Перескладання фахового вступного випробування не дозволяється.

Вступник може подати апеляцію щодо результату фахового вступного випробування лише в день оголошення результатів цього випробування.

## 5. Приклад екзаменаційного білету

Національний технічний університет України  
«Київський політехнічний інститут імені Ігоря Сікорського»

Освітній ступінь доктор філософії

Спеціальність 121 Інженерія програмного забезпечення

ЕКЗАМЕНАЦІЙНИЙ БІЛЕТ № \_\_\_\_\_

1. Діаграми компонентів та їх використання.
2. Відношення між класами (узагальнення, залежність, асоціація, агрегація, композиція) та їх виявлення. Проектування класів, відношень між класами. Пакетування клас.
3. Основні задачі протоколу IP. Формат заголовку та характеристика полів, що вирішують задачі. IP. Принципи маршрутизації IP-пакетів у складеній мережі, структура таблиць маршрутизації та команди задавання статичного маршруту до мереж.

Затверджено:

Гарант освітньої програми \_\_\_\_\_ Євгенія СУЛЕМА

## **РОЗРОБНИКИ ПРОГРАМИ:**

**Сулема Євгенія Станіславівна** – гарант освітньої програми, завідувач кафедри програмного забезпечення комп'ютерних систем, доктор технічних наук, доцент

**Гаврилко Євген Володимирович** – професор кафедри інженерії програмного забезпечення в енергетиці, доктор технічних наук, професор

**Клименко Ірина Анатоліївна** – професор кафедри обчислювальної техніки, доктор технічних наук, професор

**Новотарський Анатолій Михайлович** – професор кафедри обчислювальної техніки, доктор технічних наук, професор

**Онай Микола Володимирович** – доцент кафедри програмного забезпечення комп'ютерних систем, кандидат технічних наук, доцент

**Стіренко Сергій Григорович** – завідувач кафедри обчислювальної техніки, доктор технічних наук, професор

**Стеценко Інна Вячеславівна** – професор кафедри інформатики та програмної інженерії, доктор технічних наук, професор

**Програму рекомендовано:**

Вченою радою факультету прикладної математики

Голова вченої ради \_\_\_\_\_ Іван ДИЧКА  
протокол № 9  
від « 25 » березня 2024 р.

Вченою радою факультету інформатики та обчислювальної техніки

Голова вченої ради \_\_\_\_\_ Сергій СТИРЕНКО  
протокол № 9  
від « 25 » березня 2024 р.

Вченою радою навчально-наукового інституту атомної та теплової енергетики

Голова вченої ради \_\_\_\_\_ Євген ПИСЬМЕННИЙ  
протокол №  
від «    »            2024 р.