

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ЛЬВІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ВЕТЕРИНАРНОЇ
МЕДИЦИНИ ТА БІОТЕХНОЛОГІЙ ІМЕНІ С.З. ГЖИЦЬКОГО

ЗАТВЕРДЖУЮ

В.о. ректора університету

Іван ЦАРУБЧАК

«___» _____ 2025 р.

ПРОГРАМА

вступного **фахового** випробування для здобуття
освітнього ступеня **Бакалавр** за спеціальністю
Ф6 Інформаційні системи та технології
(ОПП «Інформаційні системи та технології»)

Програму розглянуто на засіданні
Приймальної комісії ЛНУВМБ
імені С.З.Гжицького
(протокол №6 від 09.04.2025 р.)

Львів 2025

Випробування проводяться на основі складання комплексного іспиту з наступних дисциплін: Об'єктно-орієнтоване програмування; Веб-технології і веб-дизайн; Програмування для мобільних платформ; Основи проектування інформаційних систем; Основи штучного інтелекту; Інформаційна безпека; Якість програмного забезпечення та тестування.

1. Об'єктно-орієнтоване програмування

1.1. Теоретичні основи програмування: Передумови виникнення ООП. Сучасні парадигми програмування. Відмінності процедурного, об'єктного та об'єктно-орієнтованого програмування. Основні принципи ООП. Сфери застосування ООП.

1.2. Основні поняття ООП: Класи та об'єкти. Визначення та оголошення класу. Вбудовані функції класу. Змінні класу та об'єкта. Інтерфейс класу. Реалізація класу. Способи створення об'єктів. Ідентичність об'єктів. Копіювання, ініціалізація і присвоєння об'єктів.

1.3. Принципи об'єктно-орієнтованого програмування: Принцип інкапсуляції даних. Принцип наслідування. Ієрархія об'єктів і класів. Поліморфізм. Get-тери і set-тери. Перевірка приналежності об'єкта до класу та дочірнього класу до батьківського.

1.4. Життєвий цикл об'єктів та керування пам'яттю: Тривалість життя змінних. Автоматичне і примусове очищення пам'яті. Області видимості змінних класу, об'єкта, методу. Ліквідація об'єктів. Властивості деструкторів.

1.5. Шаблони проектування (Design Patterns): Що таке шаблони проектування і навіщо вони потрібні? Класифікація шаблонів: породжувальні, структурні, поведінкові. Відмінності між шаблоном і архітектурним патерном.

1.6. Об'єктно-орієнтоване програмування та безпека: Інкапсуляція як механізм контролю доступу до даних. Безпечне використання наслідування (проблема відкритих методів). Ризики при використанні поліморфізму (наприклад, через підміну поведінки). Створення безпечного API на базі ООП.

1.7. Особливості реалізації ООП в різних мовах: Порівняння ООП у C++, Java, Python, C#. Особливості управління пам'яттю в різних мовах.

2. Веб-технології і веб-дизайн

2.1. Основи веб-розробки та дизайну — загальна методологія створення веб-сайтів: від ідеї до реалізації. Основні етапи веб-розробки, планування структури, функціональності сайту, а також сучасні тренди та принципи веб-дизайну, які визначають візуальну і користувацьку привабливість ресурсу.

2.2. Мова розмітки та стилі — базові технології побудови веб-інтерфейсу: HTML для структурування контенту і CSS для оформлення. Принципи написання валідного HTML-документу та особливості стилізації елементів за допомогою каскадних таблиць стилів.

2.3. Динамічність та клієнтське програмування — JavaScript, DOM-моделі та створенню інтерактивної поведінки на стороні клієнта. Особливості синтаксису, взаємодія зі структурою документа та об'єктно-орієнтований підхід до програмування в JavaScript.

2.4. *Серверне програмування та мови* — створення логіки, яка виконується на сервері. Використання PHP, правила написання серверного коду та принципи взаємодії між клієнтом і сервером у веб-додатках.

2.5. *Комунікація з сервером та бази даних* — протоколи обміну даними (зокрема HTTP) та роботу з базами даних через MySQL. Зв'язок між фронтендом і бекендом, збереження та обробка даних.

2.6. *Аналіз, оптимізація та безпека веб-сайтів* — методи тестування, оцінки продуктивності та оптимізації роботи веб-ресурсів. Типові загрози безпеці, шляхи їх уникнення та захист веб-додатків.

2.7. *Адаптивність та мобільна оптимізація* — техніки створення інтерфейсів, що однаково добре працюють на різних пристроях і екранах. Медіа-запити, mobile-first дизайн та перевірку відображення контенту на мобільних платформах.

2.8. *Фреймворки та бібліотеки* — використання сучасних інструментів для прискорення веб-розробки. Популярні бібліотеки JavaScript (React, Vue, Angular), CSS-фреймворки (Bootstrap, Tailwind) та їхнє застосування у проектах.

2.9. *Інструменти розробника та DevOps* — технічні інструменти, що забезпечують ефективну розробку та підтримку веб-додатків: системи контролю версій (Git), автоматизація збірки (Webpack, Gulp), а також практики CI/CD для автоматичного розгортання.

3. Програмування для мобільних платформ

3.1. *Налагодження та відлагодження мобільних додатків* — методи налагодження додатків у середовищах розробки, таких як Visual Studio, а також налаштування мобільних пристроїв Android для відлагодження. Способи ведення журналу подій і робота з емуляторами Android, їхнє конфігурування та застосування.

3.2. *Архітектура та структура мобільної платформи Android* — основні компоненти операційної системи Android, її будову та принципи роботи. Структура файлів маніфесту, які описують конфігурацію мобільних додатків і визначають їхню поведінку.

3.3. *Побудова інтерфейсу користувача в мобільних додатках* — компоновка елементів керування, засобам створення інтерфейсу, типам компоновок у платформі Android, а також особливості використання адаптерів, наприклад ListView, для відображення списків даних.

3.4. *Управління компонентами та службами додатка* — способи організації взаємодії між різними компонентами мобільного додатка, зокрема службами, а також управління фоновими задачами, що важливо для підтримки стабільної роботи і ефективного використання ресурсів.

3.5. *Безпека мобільних додатків та захист інформації* — виявлення і опис основних видів загроз інформації під час обробки на мобільних пристроях. Методи контролю доступу в мережах мобільного зв'язку і заходи для забезпечення безпеки додатків.

3.6. *Життєвий цикл мобільного додатка* — управління станами додатка на мобільних платформах, таких як запуск, пауза, відновлення і завершення роботи. Правильне управління ресурсами і збереження даних протягом різних фаз життєвого циклу.

3.7. *Робота з базами даних і збереження даних* — способи збереження інформації локально на пристрої через SQLite, Room, SharedPreferences, а також механізми синхронізації з віддаленими серверами для забезпечення актуальності даних.

3.8. *Мережеві комунікації та робота з API* — організація мережевої взаємодії мобільних додатків із зовнішніми сервісами. REST API, робота з HTTP-запитами, обробка JSON, використання WebSocket і методи підтримки стабільного з'єднання в мобільних мережах.

3.9. *Оптимізація продуктивності та управління пам'яттю* — підходи до підвищення швидкодії додатків, ефективного використання енергії, організації потоків і асинхронного програмування. Запобігання витокам пам'яті та правильне управління ресурсами для забезпечення стабільної роботи.

4. Основи проектування інформаційних систем

4.1. *Основи системного підходу та поняття інформаційної системи* — сутність системного підходу до проектування, визначення інформаційної системи та її складових, основні властивості систем і види забезпечення, а також класифікацію різновидів інформаційних систем.

4.2. *Вимоги та технічне завдання на інформаційну систему* — структура технічного завдання та вимоги, які слід враховувати при виборі засобів проектування інформаційних систем.

4.3. *Моделювання інформаційних систем* — мови та нотації присвячені елементам нотації UML, механізми доповнення діаграми IDEF0 діаграмами DFD, принципам побудови моделі IDEF0, а також функціональній моделі, бізнес-моделі процесів, моделі потоку даних і моделі життєвого циклу інформаційної системи.

4.4. *Проектування користувацького інтерфейсу* — види інтерфейсів користувача та інструментарій для їх створення.

4.5. *CASE-технології та методи проектування* — переваги і недоліки застосування CASE-технологій, а також класифікацію методів проектування за різними ознаками.

4.6. *Життєвий цикл інформаційної системи* — основні етапи розвитку інформаційної системи від початкового проектування до впровадження і підтримки.

4.7. *Аналіз вимог і збір інформації* — методи збору та формалізації вимог до системи, що є базою для її подальшого проектування.

4.8. *Архітектура інформаційних систем* — підходи до побудови архітектури, включаючи розподілені та клієнт-серверні системи, а також питання інтеграції підсистем.

4.9. *Управління проектом інформаційної системи* – планування, організацію, контроль та управління ресурсами проекту, зокрема управління ризиками, бюджетом і строками виконання.

5. Основи штучного інтелекту

5.1. *Архітектура та компоненти штучних нейронних мереж* – вхідні та вихідні нейрони, структуру штучного нейрона, а також функції активації і принципи їх побудови.

5.2. *Навчання штучних нейронних мереж* – процес навчання, призначення, особливості реалізації, класифікацію методів навчання та їх порівняння, а також переваги і недоліки практичного застосування ШНМ.

5.3. *Методи штучного інтелекту* – дерева рішень включають поняття дерева рішень, його структуру та елементи, а також алгоритми формування дерев рішень.

5.4. *Бази знань та машинне навчання* – поняття бази даних і бази знань, описують процес формування і практичного використання баз знань, а також класифікацію та підходи машинного навчання.

5.5. *Етика і застосування штучного інтелекту* – актуальні етичні питання машинного навчання і сферу використання методів ШІ з практичними прикладами.

5.6. *Алгоритми та архітектури штучного інтелекту* – різноманітні алгоритми (евристичні, генетичні, логічне програмування) і архітектурні підходи в системах ШІ.

5.7. *Обробка природної мови та комп'ютерний зір* – методи розпізнавання, аналізу і генерації природної мови, а також обробці зображень і відео.

5.8. *Оцінка ефективності моделей штучного інтелекту* – методи оцінювання якості моделей, метрики точності, валідації і узагальнення результатів.

6. Інформаційна безпека

6.1. *Теоретичні основи інформаційної безпеки* – базові поняття, визначення й концептуальні підходи до забезпечення безпеки. Сутність інформаційної безпеки держави, суспільства та особистості, її мета, основні завдання та визначення інформаційної боротьби як складової захисної діяльності.

6.2. *Загрози та моделі інформаційної безпеки* – аналіз джерел загроз, їх класифікації, умов виникнення, а також опису класичних і багаторівневих моделей інформаційної безпеки, які слугують основою для побудови систем захисту.

6.3. *Захист конфіденційної інформації* – ідентифікація конфіденційних відомостей, їх класифікації за ступенем критичності та засобах комплексного захисту. Методи виявлення й нейтралізації загроз, що стосуються витоку чи втрати даних.

6.4. *Фізичний і технічний захист інформації* – елементи мережевої безпеки персональних комп'ютерів, побудову систем фізичного захисту, включно з поняттям стримування, а також використання фізичних заходів для забезпечення безпеки на рівні підприємства.

6.5. *Правові та нормативні засади інформаційної безпеки* — основні міжнародні та національні законодавчі акти, стандарти (наприклад, ISO/IEC 27001) і нормативні документи, що регламентують політику безпеки в державних і приватних структурах.

6.6. *Організація політики інформаційної безпеки* – створення, впровадження та підтримку внутрішніх політик, інструкцій, регламентів у сфері ІБ, а також роль служб безпеки в організаційній структурі підприємства чи установи.

6.7. *Соціальна інженерія та людський фактор в ІБ* – вразливості, пов'язані з діями або помилками людей. Методи соціальної інженерії, фішинг, маніпуляції з доступом, а також способи підвищення кіберграмотності користувачів.

7. Якість програмного забезпечення та тестування

7.1. *Основи якості програмного забезпечення* охоплюють базові поняття якості програмного продукту, моделі якості та методи її контролю. Розглядається різниця між забезпеченням якості (QA) та контролем якості (QC), а також загальні підходи до оцінки якості на всіх етапах життєвого циклу програмного забезпечення.

7.2. *Методології та моделі життєвого циклу ПЗ* зосереджуються на описі життєвого циклу програмного забезпечення, типах моделей (каскадна, спіральна, V-модель тощо) і принципах Agile-маніфесту. Наведено 12 ключових принципів Agile як основи гнучкої розробки.

7.3. *Методи тестування програмного забезпечення* включають класичні підходи – «білої», «чорної» та «сірої скриньки», а також класифікацію методів тестування за критеріями доступу до внутрішньої логіки, обсягу покриття коду та рівнів абстракції.

7.4. *Типи та види тестування* охоплюють функціональне, нефункціональне, модульне, інтеграційне, системне, регресійне тестування тощо. Окрема увага приділяється тестуванню продуктивності (навантажувальному, стресовому, стабільності) та позитивному й негативному сценаріям перевірки.

7.5. *Тестова документація та тест кейси* описує структуру, правила складання та вимоги до тест-кейсів, особливості юз-кейсів і їх зв'язок з функціональними вимогами, а також підхід до написання перевіряльних сценаріїв з урахуванням критеріїв прийнятності.

7.6. *Аналіз помилок та робота з дефектами* включає класифікацію дефектів, процеси локалізації помилок, написання баг-репортів та основні принципи взаємодії між тестувальниками й розробниками щодо усунення помилок.

7.7. *Автоматизоване тестування* — ця рубрика (додаткова) охоплює поняття автоматизації, інструменти (наприклад, Selenium, JUnit, TestNG), області застосування, переваги та обмеження автоматизованого тестування, а також принципи побудови тест-скриптів і підтримки автотестів.

7.8. *Тестування безпеки програмного забезпечення* — ще одна важлива рубрика, яка охоплює вразливості ПЗ, принципи тестування на проникнення (penetration testing), інструменти та методи виявлення загроз, захисту даних, автентифікації та авторизації.

7.9. *Неперервна інтеграція та DevOps у тестуванні* — розглядаються сучасні практики CI/CD (Continuous Integration / Continuous Delivery), роль тестування в процесі автоматизованої збірки, розгортання та релізу, а також взаємодія команд у межах DevOps-культури.

Список рекомендованої літератури

Об'єктно-орієнтоване програмування

1. Портал об'єктно-орієнтованого програмування: <http://oop.in.ua/tag/FAQ/>

2. Книжки з програмування: <https://dou.ua/lenta/articles/programming-books/>

3. Українська технічна література. Програмування: <https://ukrtechlibrary.wordpress.com/tag/програмування/>

4. Вступ до Python 3: <https://www.learnpython.org/>

Веб-технології і веб-дизайн

1. Трофименко О.Г., Козін О.Б., Задерейко О.В., Плачінда О.Є. Веб-технології та веб-дизайн : навч. посібник, Одеса:Фенікс, 2019.-284 с.

2. Соломін А.В. Веб-орієнтована розробка програмного забезпечення. К.:КПІ ім.Сікорського, 2018. 131с.

3. Бородкіна І.Л., Бородкін Г.О. Web-технології та web-дизайн. Ліра-К, 2020. 212 с.

4. Пасічник В.В., Пасічник О.В., Угрин Д.І. Веб-технології. Львів: "Магнолія 2006", 2018. 336 с.

5. Мельник Р. Програмування веб-застосувань (фронт-енд та бек-енд). Львів: НУ «Львівська політехніка», 2018. 248с.

Програмування для мобільних платформ

1. Аналіз методів і технологій розробки мобільних додатків для платформи Android : навч. посіб. / О. В. Шматко, А. О. Поляков, В. М. Федорченко. Харків : НТУ «ХПІ», 2018. 284 с.

2. Програмне забезпечення мобільних пристроїв: навчальний посібник. Львів: Видавництво «Новий Світ-2000» 2020. 218 с.

3. Програмування мобільних пристроїв: навчальний посібник для дистанційного навчання / К.Т. Кузьма. Миколаїв: СПД Румянцева Г. В., 2021. 128 с.

Основи проектування інформаційних систем

1. Литвин В.В., Пасічник В.В., Шаховська Н.Б. Проектування інформаційних систем. Навчальний посібник (затв. МОН України). Львів: Магнолія 2006, 2021. 380 с.

2. Авраменко В.С., Авраменко А.С. Проектування інформаційних систем: навчальний посібник. Черкаси: Черкаський національний університет ім. Б. Хмельницького, 2017. 434 с.

3. Недашківський О.М.. Планування та проектування інформаційних систем. Київ, 2014. 215 с.

4. Проектування інформаційних систем: Загальні питання теорії проектування ІС (конспект лекцій) [Електронний ресурс]: навч. посіб. для студ. спеціальності 122 «Комп'ютерні науки» / КПІ ім. Ігоря Сікорського; уклад.: О. С. Коваленко, Л. М. Добровська. Київ : КПІ ім. Ігоря Сікорського, 2020. 192 с.

Основи штучного інтелекту

1. Савченко А.С., Синельников О.О. Методи та системи штучного інтелекту: навчальний посібник, Київ : НАУ, 2017, 190 с.

2. Лубко В.Д., Шаров С.В. Методи та системи штучного інтелекту: навчальний посібник, Мелітополь : ФОП Однорог Т.В., 2019, 264 с.

3. Коцовський В.М., Методи та системи штучного інтелекту: конспект лекцій, Ужгород : Ужгородський національний університет, 2016, 76 с.

4. Субботін С.О., Олійник А.О. Нейронні мережі: навчальний посібник, Запоріжжя : ЗНТУ, 2014, 132 с.

Інформаційна безпека

1. ДСТУ 3396.0-96. Захист інформації. Технічний захист інформації. Основні положення.

2. Закон України “Про захист інформації в інформаційно-телекомунікаційних системах” від 05.07.1994 // Відомості Верховної Ради України, 1994, № 31.

3. Захист інформації в інформаційно-комунікаційних системах. Навчальний посібник для курсантів ВНЗ МВС України / О.В. Рибальський, В.Г. Хахановський, В.А. Кудінов, В.М. Смаглюк. К.: Вид. Національної академії внутріш. справ, 2013. 118 с.

4. Основи інформаційної безпеки. Підручник для курсантів ВНЗ МВС України / Рибальський О.В., Смаглюк В.М., Хахановський В.Г. К.: НАВС, 2013. 255 с.

5. Технології захисту інформації: підручник для студ. спеціальності 122 «Комп'ютерні науки», спеціалізацій «Інформаційні технології моніторингу довкілля», «Геометричне моделювання в інформаційних системах» / Ю. А. Тарнавський; КПІ ім. Ігоря Сікорського. Київ: КПІ ім. Ігоря Сікорського, 2018. 162 с.

Якість програмного забезпечення та тестування

1. Алексенко О.В. Технології програмування та створення програмних продуктів: конспект лекцій / О.В. Алексенко. Суми: Сумський державний університет, 2013. 133 с.

2. Бевз О.М., Папінов В.М., Скидан Ю.А. Проектування програмних засобів систем управління. URL: https://web.posibnyky.vntu.edu.ua/fksa/4bevz_proektuvannya_programnyh_zasobiv_system_upravlinnya/2.htm

3. Козак О.Л. Опорний конспект лекцій з курсу «Якість програмного забезпечення та тестування» для студентів напрямку підготовки «Програмна інженерія» / О.Л. Козак. Тернопіль, 2012. 72 с.

4. Якість програмного забезпечення та тестування: базовий курс. Навчальний посібник / За ред. Крепич С.Я., Співак І.Я. / для бакалаврів галузі знань 12 «Інформаційні технології» спеціальності 121 «Інженерія програмного забезпечення». Тернопіль: ФОП Паляниця В.А., 2020. 478 с.

КРИТЕРІЇ ОЦІНЮВАННЯ ЗНАНЬ ВСТУПНИКІВ

Оцінювання вступного фахового випробування для здобуття освітнього ступеня Бакалавр проводиться за 200-бальною шкалою (від 100 до 200 балів). Позитивна оцінка становить не менше 100 балів. Екзаменаційний білет містить чотири питання, кожне з яких оцінюється по 25 балів максимально.

За повну та правильну відповідь на всі запитання вступник може набрати максимально 200 балів (за 200-бальною шкалою).

При цьому до участі в конкурсному відборі допускаються особи, які отримали за результатами вступного фахового випробування не менше 100 балів.

Критерії оцінювання відповідей на питання фахового вступного випробування:

- відповідь у 90–100% від кількості балів оцінюється, якщо вступник у повному обсязі розкрив зміст питання; здатен формувати висновки й узагальнення, вільно оперувати фактами та відомостями; правильно розв'язав завдання;
- відповідь у 70–80% від кількості балів оцінюється, якщо вступник достатньо повно розкрив зміст відповіді, але при викладанні деяких аспектів не вистачає достатньої глибини та аргументації, допускаються при цьому окремі несуттєві неточності та незначні помилки; правильно розв'язав завдання, але допустив незначні неточності;
- відповідь у 50–60% від кількості балів оцінюється, якщо вступник у цілому розкрив основний зміст питання, але без обґрунтування та аргументації, допускаючи при цьому окремі суттєві неточності та помилки; завдання розв'язав не повністю;
- відповідь до 50% від кількості балів оцінюється, якщо вступник недостатньо розкрив зміст теоретичних питань і практичних завдань, допускаючи при цьому суттєві неточності; завдання розв'язав частково або неправильно.