

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ  
БУДІВНИЦТВА І АРХІТЕКТУРИ

магістр

Факультет:

Автоматизації і інформаційних технологій

«Затверджую»  
Голова приймальної комісії  
Ректор  
П. М. Куліков



**ПРОГРАМА**

вступних фахових випробувань  
до вступу на навчання для отримання  
ступеня магістра зі спеціальності  
**121 «ІНЖЕНЕРІЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ»**  
галузі знань 12 «Інформаційні технології»

Затверджено на засіданні  
приймальної комісії, протокол  
№ 4 від « 18 » лютого 2019 р.

## 1. ЗАГАЛЬНІ ПОЛОЖЕННЯ

Вступні випробування проводяться фаховою атестаційною комісією для осіб, які закінчили ВНЗ та отримали диплом за освітнім ступенем бакалавра (освітньо-кваліфікаційним рівнем спеціаліста) і вступають на спеціальність 121. «Інженерія програмного забезпечення». Бажаючі навчатися для отримання ступеня магістра складають контрольні заходи у формі тестування з фахових дисциплін «Об'єктно-орієнтоване програмування» і «Організація баз даних та знань».

Другий (магістерський) освітній рівень є професійно-орієнтованим і обов'язковим для продовження навчання за однією зі спеціальностей в аспірантурі.

Студенти спеціальності 121. «Інженерія програмного забезпечення» опановують знання, уміння та навички, які дозволяють їм: створення програмного забезпечення, у тому числі у сфері інтернет технологій, мережевого і комунікаційного програмного забезпечення; проектування інтерактивних інтерфейсів користувача; проектування і розробка баз даних; тестування та забезпечення якості програмного забезпечення; аналіз проектів та програм готових систем, аналіз потреб та проблем користувачів, пошук найоптимальніших рішень; розробка, виготовлення, постачання та документування програмного забезпечення на замовлення користувачів; надання консультацій у сфері проектування і розробки програмного забезпечення.

Місцем роботи випускників можуть бути організації, які займаються розробкою та супроводом програмного забезпечення, незалежні агенції з тестування програмного забезпечення.

Після закінчення навчання випускники отримують диплом державного зразка та кваліфікацію – «професіонал в галузі з інженерії програмного забезпечення».

## 2. ЗМІСТ ПРОГРАМИ ВСТУПНОГО ВИПРОБУВАННЯ

### 2.1. Дисципліна «Об'єктно-орієнтоване програмування»

- 2.1.1 Класифікація мов програмування.
- 2.1.2 Поняття об'єкту та його порівняння зі структурами даних та алгоритмів в мові C.
- 2.1.3 Основні переваги визначення методів у мові C#.
- 2.1.4 Характеристика каркасу Framework.Net .
- 2.1.5 Створення багаторівневої ієрархії класів у мові C#.
- 2.1.6 Об'єктно-орієнтований аналіз та проектування .
- 2.1.7 Система типів мови C#.
- 2.1.8 Оператори new та delete для керування пам'яттю.
- 2.1.9 Перевантаження операторів true false у мові C#
- 2.1.10 Перевантаження бінарних операторів у мові C#
- 2.1.11 Перевантаження унарних операторів у мові C#.
- 2.1.12 Основні поняття ООП. Класи та об'єкти.
- 2.1.13 Клас та екземпляр. Найпростіший синтаксис означення класу.
- 2.1.14 Специфікатори доступу public, protected та private.
- 2.1.15 Застосування абстрактних класів у мові C#.
- 2.1.16 Рядки у мові C# основні концепції роботи з ними.
- 2.1.17 Використання класу StringBuilder для побудови рядків у мові C#
- 2.1.18 Поняття інкапсуляції. Поняття про захист внутрішніх даних об'єкту.
- 2.1.19 Метод як канал доступу до внутрішніх даних. Модель об'єкта як чорна скринька.
- 2.1.20 Конструктори, деструктори, їх роль та призначення.
- 2.1.21 Динамічне виділення пам'яті для об'єкту: виклик конструктора з оператору new.
- 2.1.22 Конструктори по замовчуванню та з параметрами. Особливості ініціалізації членів-даних.
- 2.1.23 Наслідування. Механізм наслідування членів-даних та методів.
- 2.1.24 Доступ до членів класу та наслідування у мові C#
- 2.1.25 Відношення клас-підклас та його зв'язок з відношенням абстрактне-конкретне. Сумісність типів знизу вгору.
- 2.1.26 Особливості виклику конструктора базового класу з конструктору надкласу. Порядок виклику конструкторів та деструкторів для об'єктів похідних класів.
- 2.1.27 Ієрархія класів. Множинне наслідування та його проблеми.

- 2.1.28 Поліморфізм та віртуальні функції. Поняття оголошеного та фактичного типу.
- 2.1.29 Механізм виклику віртуальної функції.
- 2.1.30 Дружні функції, окремі класи та класи в цілому, обхід механізмів захисту членів класу.
- 2.1.31 Переваги та недоліки використання механізму дружності в програмах з об'єктно-орієнтованою композицією.
- 2.1.32 Перевантаження операторів функціями та методами.
- 2.1.33 Віртуальні методи у мові C#.
- 2.1.34 Класи потоків введення-виведення. Управління форматом, модифікатори.
- 2.1.35 Шаблони функцій та класів.
- 2.1.36 Організація захищеного доступу у мові C#.
- 2.1.37 Перевантаження логічних операторів у мові C#.
- 2.1.38 Обробка виняткових ситуацій. Поняття про виняткову ситуацію.
- 2.1.39 Оператори try, throw і catch.
- 2.1.40 Порядок збудження та перехоплення виняткової ситуації.
- 2.1.41 Класи колекцій стандартної бібліотеки: послідовні контейнери.
- 2.1.42 Класи колекцій стандартної бібліотеки: асоціативні контейнери.
- 2.1.43 Основи стандартної бібліотеки: типи ітераторів та операції над ними.
- 2.1.44 Основи стандартної бібліотеки: узагальненні алгоритми. Види узагальнених алгоритмів. Приклад.
- 2.1.45 Основи стандартної бібліотеки: поняття функора. Адаптери функорів.
- 2.1.46 Графічна мова UML. Діаграма прецедентів.
- 2.1.47 Графічна мова UML. Діаграма класів.
- 2.1.48 Графічна мова UML. Діаграма використання.
- 2.1.49 Графічна мова UML. Діаграма послідовностей.
- 2.1.50 Графічна мова UML. Діаграма діяльності.

## ЛІТЕРАТУРА ДЛЯ ПІДГОТОВКИ

1. Федоров А.Г. JavaScript для всех [Текст]: учеб. / –М.: КомпьютерПресс, 2001. – 384 с.
2. Шапошников И. Интернет-программирование [Текст]: учеб. / – СПб: BHV, 2000. – 224 с.
3. Эндриус Г. Основы многопоточного, параллельного и распределенного программирования – М.: Вильямс – 512 с.

4. Прата С. Язык программирования Си. Лекции, упражнения. / С.Прата.- М.: Вильямс, 2006.- 260 с.
5. 3. Хабібулін . І.Ш. Программирование С++: Пер. з англ. – 3-е вид. – СПб.: БХВ-Пітербург, 2006. – 512 с.
6. 4. Повний довідник по С++, 4-е видання.: Перекл.з англ. – М.: Видавничий дім «Вільямс», 2011. – 800 с.
7. 5. Об'єктно-орієнтоване програмування в С++. Класика Computer Science. 4-е вид. – СПб.: Пітер, 2013. – 928 с.

## 2.2. Дисципліна «Організація баз даних і знань».

- 2.2.1 Основні визначення та поняття бази даних і знань.
- 2.2.2 Поняття системи управління базою даних.
- 2.2.3 Основні поняття та визначення реляційної бази даних.
- 2.2.4 Поняття об'єктно-орієнтованої бази даних.
- 2.2.5 Представлення концептуальної моделі бази даних.
- 2.2.6 Функції виконання фактографічної інформаційної системи.
- 2.2.7 Моделі даних, що реалізує сучасні СУБД.
- 2.2.8 Поняття та визначення незалежність даних.
- 2.2.9 Вигляд представлення запиту в реляційному обчисленні зі змінними кортежами.
- 2.2.10 Проблеми, для вирішення яких необхідно виконувати нормалізацію бази даних.
- 2.2.11 Визначення і поняття першої нормальної форми.
- 2.2.12 Визначення і поняття другої нормальної форми..
- 2.2.13 Визначення і поняття третьої нормальної форми..
- 2.2.14 Визначення поняття “цілісність бази даних”.
- 2.2.15 Визначення поняття «відношення».
- 2.2.16 Визначення поняття «кортеж».
- 2.2.17 Визначення поняття “домен”.
- 2.2.18 Поняття «ключ» в реляційній базі даних.
- 2.2.19 Визначення поняття «зовнішній ключ».
- 2.2.20 Визначення поняття «реплікація».
- 2.2.21 Визначення поняття «тригер».
- 2.2.22 Визначення поняття “транзакція”.
- 2.2.23 Використання метадані в базах даних.

- 2.2.24 Особливості архітектури файл-сервер.
- 2.2.25 Реалізація, що дозволяє зробити індекси.
- 2.2.26 Дії, що передбаченні для забезпечення безпеки даних в базі даних.
- 2.2.27 Функції виконання представлення (view).
- 2.2.28 Архітектура програмно-технічних засобів розподілених СУБД.
- 2.2.29 Функції виконання сервер застосувань.
- 2.2.30 Використання технології, що передбачає реплікацію.
- 2.2.31 Використання технології, що передбачає фрагментацію.
- 2.2.32 Рівні ієрархії пристроїв пам'яті бази даних.
- 2.2.33 Використання кеш-пам'ять для потреб.
- 2.2.34 Визначення та поняття дамп бази даних.
- 2.2.35 Застосування мето-поле в базах даних.
- 2.2.36 Застосування В-дерева в базах даних.
- 2.2.37 Визначення "ADO".
- 2.2.38 Визначення "OLE DB".
- 2.2.39 Представлення інтерфейсу ISAPI/NSAPI.
- 2.2.40 Визначення "QBE".
- 2.2.41 Визначення типу мов програмування, що відносяться до мови SQL.
- 2.2.42 Вигляд, що має правильний синтаксис для виключення пустих значень з таблиці.
- 2.2.43 Команди, які включають мову запитів до бази даних.
- 2.2.44 Команди, які включають мову маніпулювання з даними DML.
- 2.2.45 Команди, які на мові SQL застосовуються для визначення обмежень полів в базі даних.
- 2.2.46 Визначення "ACID".
- 2.2.47 Визначення і поняття інкрементного поля.
- 2.2.48 Представлення програми – клієнт.
- 2.2.49 Правила перетворення ER-діаграм, які перетворюються в логічні моделі реляційної бази даних.
- 2.2.50 Правильність складання SQL-запитів до бази даних.
- 2.2.51 Правильність застосування команд SQL по створенню бази даних.
- 2.2.52 Правильність приведення логічної схеми бази даних до третьої нормальної форми.
- 2.2.53 Представлення "CORBA".
- 2.2.54 Класична архітектура бази даних, яка включає запропоновану ANSI.
- 2.2.55 Основні поняття та визначення сховище даних.

- 2.2.56 Рівні ізоляції транзакцій і в чому їх особливості.
- 2.2.57 Головні властивості розподіленої бази даних.
- 2.2.58 Моделі або модель, яка підтримує об'єктно-реляційну базу даних.
- 2.2.59 Поняття хешування і для чого воно використовується.
- 2.2.60 Визначення репозиторій.
- 2.2.61 Функції, що виконують серіалізацію транзакцій.
- 2.2.62 Функції, що виконують файловий сервер.
- 2.2.63 Поняття, визначення "BLOB".
- 2.2.64 Журнал реєстрації транзакцій, принцип використання.

### ЛІТЕРАТУРА ДЛЯ ПІДГОТОВКИ

1. Єрьоміна Н.В. Проектування баз даних : Навч. посібник/ Київськ. наук. економічн. ун-т – К., 1998. – 205 с.
2. Базы данных: модели, разработка, реализация: Учебник/ Е.С. Карпов – СПб: Питер, 2001. – 304 с.
3. Проектування баз даних у САПР просторових конструкцій : Навч. посібник /В.В. Лажечніков, О.Л. Синявський, В.І. Тартачник; Київськ. інж.-буд. ін-т – К.: НМК ВО, 1993. – 211 с.
4. Конноли Т., Бегг К., Страган А. Базы данных: проектирование, реализация и сопровождение. Теория и практика. :Учеб. пособ.- М.: "Вильямс", 2000. - 1120 с.
5. Архангельський А.Я. Программирование в Delphi7.-М.: Нолидж, 2006. -1124 с.
6. Харрингтон Д. Проектирование объектно-ориентированных баз данных: Пер. с англ. – М.: ДМК Пресс, 2001. - 272 с.
7. Хендерсон Кен. Delphi 3 и системы клиент/сервер. Руководство разработчика. - Пер. с англ. - К.:Диалектика, 1997. – 736 с.
8. Шумаков П.В., Фаронов В.В. Delphi 6. Руководство разработчика баз данных.-М.: Нолидж, 2005. -640 с.
9. Microsoft Windows 2000 Server: Русская версия\ А.Андреев, Е. Беззубов, Е. Емельянов и др.; Под общ. ред. А. Чекмарева, Д. Вишнякова. – СПб.: БХВ Санкт-Петербург, 2000. – 910 с.

### 3. СТРУКТУРА ЕКЗАМЕНАЦІЙНОГО БІЛЕТА. КРИТЕРІЇ ОЦІНЮВАННЯ

Оцінювання знань вступників на вступному випробуванні здійснюється за шкалою від 60 до 100 балів по кожній із зазначених дисциплін. Вступне випробування включає тестові завдання з дисциплін, кожне з яких налічує 20 питань. Кожне питання має чотири відповіді позначені літерами (цифрами), одна з яких вірна. Вступник обирає правильну відповідь до тестового питання та позначає її відповідною літерою (цифрою) напроти номера питання у стовпчику «відповідь» бланка-відповіді. Якщо вступник вирішив виправити відповідь на питання, то має внести зміну літерою (цифрою) у стовпчик «виправлена відповідь». Прийнятим до оцінювання буде запис внесений у стовпчик «виправлена відповідь». Викреслювати відповіді (літери, цифри) не дозволяється. Кожна правильна відповідь на питання оцінюється в 1 (один) бал, неправильна відповідь – 0 (нуль) балів і за таблицею переведення визначається конкурсна оцінка з фахового випробування.

Таблиця переведення результатів випробування (тестування) з фахової дисципліни із 20-ти бальної у 100-бальну шкалу оцінювання знань

Кількість вірних відповідей на запитання тестового завдання	Кількість балів за шкалою ECTS
0...5	0 (незадовільно – контрольний захід нескладений)
6	60
7	61
8	63
9	65
10	68
11	71
12	74
13	77
14	80
15	83
16	86
17	89
18	92
19	96
20	100



Якщо вступник не склав контрольний захід хоча б по одній з дисциплін, вважається що він не пройшов фахове випробування та втрачає право брати участь в конкурсі на навчання для отримання ступеня магістра.

За результатами вступного випробовування визначається сумарна кількість балів з зазначених дисциплін, на підставі якої фахова атестаційна комісія вносить на розгляд приймальної комісії університету затвердження результатів фахового випробування.

Голова фахової атестаційної комісії



I.B. Русан