

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
МІЖНАРОДНИЙ ГУМАНІТАРНИЙ УНІВЕРСИТЕТ  
ФАКУЛЬТЕТ КІБЕРБЕЗПЕКИ, ПРОГРАМНОЇ ІНЖЕНЕРІЇ ТА  
КОМП'ЮТЕРНИХ НАУК  
КАФЕДРА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

«ЗАТВЕРДЖУЮ»

Голова приймальної комісії

Міжнародного гуманітарного  
університету

Ректор



К.В. Громовенко

2023 р

**ПРОГРАМА**

додаткового фахового вступного випробування  
для здобуття другого (магістерського) рівня вищої освіти  
на основі раніше здобутого першого (бакалаврського) рівня, другого  
(магістерського) рівня освіти або освітньо-кваліфікаційного рівня  
спеціаліста

ГАЛУЗЬ ЗНАНЬ

12 Інформаційні технології

(шифр та назва галузі знань)

СПЕЦІАЛЬНІСТЬ

121 Інженерія програмного забезпечення

(шифр та назва спеціальності)

ОСВІТНЯ ПРОГРАМА

Інженерія програмного забезпечення

(назва освітньої програми)

**Розглянуто та схвалено:**

на засіданні кафедри інформаційних технологій

Протокол № 9 від 28 березня 2023 р.

**Розглянуто та схвалено:**

на засіданні Вченої ради Міжнародного гуманітарного університету

Протокол № 6/1 від 7 квітня 2023 р.

*Введено в дію Наказом Міжнародного гуманітарного університету  
№ 574а від 07.04.2023*

Одеса 2023

Програма додаткового фахового вступного випробовування для здобуття другого (магістерського) рівня вищої освіти зі спеціальності 121 Інженерія програмного забезпечення.

Розробник:

Данилець Є.В., к.т.н., доцент кафедри інформаційних технологій  
Міжнародного гуманітарного університету;

Завідувачка кафедри  
інформаційних технологій,  
к.т.н., доцент



Т.І. Григор'єва

Програма розглянута та схвалена на засіданні кафедри інформаційних технологій, протокол № 9 від 28 березня 2023 р.



## ПОЯСНЮВАЛЬНА ЗАПИСКА

Особи, які раніше здобули перший (бакалаврський) рівень, другий (магістерський) рівень освіти або освітньо-кваліфікаційний рівень спеціаліста з іншої спеціальності, обов'язково складають додаткове фахове вступне випробування. Абітурієнт повинен на додатковому фаховому вступному випробуванні продемонструвати фундаментальні та професійно-орієнтовані уміння та знання передбачені для спеціальності 121 Інженерія програмного забезпечення.

На додаткове фахове вступне випробування виносяться питання з наступних дисциплін:

1. Алгоритми та структури даних.
2. Аналіз вимог до програмного забезпечення.
3. Архітектура та проектування програмного забезпечення.
4. Організація баз даних та знань.
5. Конструювання програмного забезпечення.
6. Веб-технології та веб-дизайн.
7. Проектування користувальницьких інтерфейсів.
8. Менеджмент проектів програмного забезпечення.
9. Моделювання та аналіз програмного забезпечення.
10. Об'єктно-орієнтоване програмування.
11. Якість програмного забезпечення та тестування.

Організація додаткового фахового вступного випробування здійснюється відповідно до Правил прийому до Міжнародного гуманітарного університету.

### **МЕТА ДОДАТКОВОГО ФАХОВОГО ВСТУПНОГО ВИПРОБУВАННЯ**

Визначення рівня підготовки абітурієнтів з метою проведення конкурсного відбору для навчання в Міжнародному гуманітарному університеті (далі: Університет) зі спеціальності 121 Інженерія програмного забезпечення.

### **ФОРМА ДОДАТКОВОГО ФАХОВОГО ВСТУПНОГО ВИПРОБУВАННЯ**

Згідно з чинним «Порядком прийому до Міжнародного гуманітарного університету у 2023 році», для охочих продовжити навчання за другим (магістерським) рівнем вищої освіти, які раніше здобули перший (бакалаврський) рівень, другий (магістерський) рівень освіти або освітньо-кваліфікаційний рівень



спеціаліста з іншої спеціальності, передбачено обов'язкове складання додаткового фахового вступного випробування, який може проводитися очно та/або дистанційно. Нижче наведена структура даного випробування та навчальні матеріали, які рекомендовані для опрацювання в ході підготовки до нього. (Додаток 1).

1. Додаткове фахове вступне випробування проводиться у вигляді тестового завдання. Абітурієнт відповідає на двадцять тестових завдань, кожне з яких має 4 варіанти відповіді. Вступнику необхідно вибрати одну правильну відповідь з кожного тестового завдання. Питання взято з відповідних навчальних програм дисциплін «Алгоритми та структури даних», «Аналіз вимог до програмного забезпечення», «Архітектура та проектування програмного забезпечення», «Організація баз даних та знань», «Конструювання програмного забезпечення», «Веб-технології та веб-дизайн», «Проектування користувальницьких інтерфейсів», «Менеджмент проектів програмного забезпечення», «Моделювання та аналіз програмного забезпечення», «Об'єктно-орієнтоване програмування», «Якість програмного забезпечення та тестування», відповідно до програм підготовки бакалаврів, спеціалістів або магістрів зі спеціальності 121 Інженерія програмного забезпечення.

2. Перелік запитань, покладених в основу вступного іспиту з фахових дисциплін, наведено в Додатку 1.

## КРИТЕРІЇ ОЦІНЮВАННЯ

Додаткове фахове вступне випробування проводиться у вигляді тестового завдання. Тестове завдання складається з 20 питань, кожне з яких оцінюється в 10 балів.

При оцінюванні знань абітурієнта, вихідними критеріями є такі:

- кожна вірна відповідь на тестове завдання оцінюється у 10 балів;
- оцінку «відмінно» абітурієнт отримує, якщо він набрав 180 або 190, або 200 балів;
- оцінку «добре» абітурієнт отримує, якщо він набрав 150 або 160, або 170 балів;
- оцінку «задовільно» абітурієнт отримує, якщо він набрав 100 або 110, або 120, або 130, або 140 балів;
- оцінку «незадовільно» абітурієнт отримує, якщо він набрав менше ніж 100 балів.



Перелік питань для додаткового фахового вступного випробування для осіб, що виявили бажання продовжити навчання для здобуття другого (магістерського) рівня вищої освіти зі спеціальності 121 Інженерія програмного забезпечення

1. Для чого використовуються псевдо елементи і що таке псевдо-елементи CSS? Наведіть приклади.
2. Класифікація вимог. Визначення поняття вимоги.
3. Під час конструювання програмного забезпечення який відсоток становить процес розв'язання задачі.
4. Які HTML-елементи використовуються для побудови структури таблиці? Наведіть приклади.
5. Переваги та недоліки структурного та об'єктно-орієнтованого моделювання.
6. Принципи структурного та об'єктно-орієнтованого моделювання.
7. Методології структурного моделювання: ER, IDEF0, DFD.
8. Діаграма варіантів застосування UML (визначення та призначення).  
Основні елементи, правила створення діаграми. Потoki подій.
9. Методи і засоби перевірки вимог.
10. Ключові архітектурні принципи.
11. Дизайн рівневої структури.
12. Аналіз вимог. Роль глосарію.
13. Джерела вимог. Стратегії виявлення вимог.
14. Документування вимог.
15. Документування архітектури ПЗ.
16. Шаблони проектування.
17. Архітектурні стилі.
18. Система керування базами даних. Призначення. Основні характеристики.
19. Операції реляційної алгебри.
20. Моделі даних.
21. Реляційна модель даних.
22. Мова SQL: оператори визначення даних та оператори маніпулювання даними.
23. Основні поняття і призначення. Нормальні форми.
24. Які різновиди моделей життєвого циклу програмного забезпечення Вам відомі?



25. Основні можливості конфігураційних мов конструювання.
26. Основні можливості інструментальних мов конструювання.
27. Що таке побудова, перевірка, розгортання?
28. Перелічить основні техніки відлагодження програмного забезпечення.
29. Яким чином можна оголосити функцію в JavaScript? Наведіть приклади.
30. Які комбінації селекторів ви знаєте? Опишіть їх.
31. Яким чином можна підключити зовнішню таблицю стилів в html-документі?
32. Класифікація функціональних характеристик продукту на основі цінності для цільової аудиторії за допомогою моделі Кано.
33. Методика «Persona» в задачі розробки профайлу користувача.
34. Використання User Story в проектуванні інтерфейсу програмного забезпечення.
35. Сценарії в проектуванні інтерфейсу програмного забезпечення.
36. Перерахуйте моделі життєвого циклу проекту. Наведіть переваги та недоліки кожної моделі.
37. Основні проектні документи: їхні змісти та цілі.
38. Принципи та підходи до розроблення ієрархічної структури робіт.
39. Методи оцінювання тривалості робіт проекту.
40. Методи ідентифікації та оцінювання проектних ризиків.
41. Діаграма класів UML (призначення, застосування). Елементи діаграми класів UML. Типи відношень між класами. Стереотипи діаграми класів.
42. Діаграми послідовності UML (визначення та призначення). Елементи діаграми послідовності.
43. Клас та його члени. Об'єкти. Конструктори, властивості, методи, індексатори. Перевантаження операцій. Навести приклади.
44. Статичні члени класів. Навести приклади.
45. Спадкоємність, поліморфізм. Навести приклади ієрархії класів та поліморфізма.
46. Як називається процес розбиття однієї складної задачі на кілька простих підзадач?
47. Абстрактні класи і інтерфейси. Навести приклади.
48. Що таке стійкість програмного забезпечення.
49. Задачі валідації та верифікації.
50. Поняття якості програмного забезпечення. Тестовий випадок (Test Case): визначення, структура, види тестових випадків.
51. Що виконує користувач на етапі користування.



## ПЕРЕЛІК РЕКОМЕНДОВАНОЇ ЛІТЕРАТУРИ

1. Коротєєва Т. О. Алгоритми та структури даних. Навчальний посібник / Т. О. Коротєєва. Львів : Видавництво Львівської політехніки, 2014. – 280 с.
2. Алгоритми та структури даних: Навчальний посібник для студентів технічних спеціальностей денної та заочної форми навчання. – Кропивницький: Видавець – Лисенко В.Ф., 2019. – 156 с.
3. Алгоритми, дані і структури. [Текст], навч. посіб. / В.М. Ільман, О.П. Іванов, Л.О. Панік. Дніпропет. нац. ун-т залізн. трансп.ім. акад. В. Лазаряна. – Дніпро, 2019. – 134 с.
4. Ю. Грицюк. Аналіз вимог до програмного забезпечення. Львівська політехніка. – 2018. – 418 с.
5. L. Bass, P. Clements, R. Kazman. Software Architecture in Practice. 3-d edition, Addison-Wesley, 2012. – 600 p.
6. М. Фаулер. Шаблоны корпоративных приложений. Исправленное издание. — М. : Вильямс, 2016. — 548 с.
7. R. Pressman, B. Maxim. Software Engineering: A Practitioner's Approach. 9th Edition. McGraw Hill. 2020.
8. T. Nield, Getting Started with SQL: A Hands-On Approach for Beginners / O'Reilly Media, 2016. – 134 p.
9. R.O. Obe, L.S. Hsu, PostgreSQL: Up and Running: A Practical Guide to the Advanced Open Source Database, 3rd edition / O'Reilly Media, 2017. – 314 p.
10. C.J. Date, Database Design and Relational Theory: Normal Forms and All That Jazz, 2nd ed. edition / Apress, 2019. – 470 p.
11. Малахов Є.В., Проектування баз даних та їх реалізація засобами стандартного SQL та PostgreSQL: Навч. посіб. для студ. вищих навч. закладів / Є.В. Малахов, О.А. Блажко, М.Г. Глава // Одеса: ВМВ, 2014. – 248 с.
12. Г. Гайна, Основи проектування баз даних. Навчальний посібник / Кондор, 2018. – 204 с.
13. Refactoring: Improving the Design of Existing Code, by Martin Fowler Publisher: Addison-Wesley Professional; 1 edition, and 2 edition, 2019. URL: [https://www.csie.ntu.edu.tw/~r95004/Refactoring\\_improving\\_the\\_design\\_of\\_existing\\_code.pdf](https://www.csie.ntu.edu.tw/~r95004/Refactoring_improving_the_design_of_existing_code.pdf)
14. Цеслів О.В. WEB-програмування : навч. посібник / О.В. Цеслів ; М-во освіти і науки, молоді та спорту України, Нац. техн. ун-т України “Київ. політехн. ін-т”. – Київ : НТУУ “КПІ”, 2011. – 296 с.

15. Пасічник В.В. Веб-дизайн / В.В. Пасічник, О.В.Пасічник // Львів: Магнолія, 2018. – 520 с.
16. Haverbeke M. Eloquent JavaScript, 3rd Edition – 2018. – 435 p.
17. Пасічник В.В. Веб-технології / В.В. Пасічник, О.В.Пасічник // Львів: Магнолія, 2019. – 336 с.
18. Robbins N. Learning Web Design: A Beginner's Guide to HTML, CSS, JavaScript, and Web Graphics, Fifth Edition / Jennifer Niederst Robbins // Shroff/O'Reilly, 2018. – 812 p.
19. Blaskovics B., Deák C., Varga A. K. Chapters from the academic aspect of project management. Research and Teaching Methodologies Volume III. PMUni, 2019. 192 p.
20. Nicholas J. M., Steyn H. Project management for business engineering and technology: 5rd ed. Routledge: Taylor & Francis Group, 2017. 1389 p.
21. Ноздріна Л. В., Ящук В. І., Полотай О. І. Управління проектами: підручник. К.: Центр учбової літератури, 2015. – 432 с.
22. Яковенко А. В., Коновал О. О. Основи програмування: методичні вказівки до виконання комп'ютерних практикумів з дисципліни «Управління ІТ-проектами». Управління ІТ-проектами. – К.: НТУУ «КПІ ім. І. Сікорського», 2017. – 47 с.
23. Авраменко В.С., Авраменко А.С. Проектування інформаційних систем: навчальний посібник / В.С. Авраменко, А.С. Авраменко. – Черкаси: Черкаський національний університет ім. Б. Хмельницького, 2017.
24. Бублик В.В. Об'єктно-орієнтоване програмування. Підручник. К.: ІТ книга, 2015. – 624 с.: іл.
25. Об'єктно-орієнтоване програмування. Частина 1. Основи об'єктно-орієнтованого програмування на мові С#: Навчальний посібник. / Д.В. Настенко, А.Б. Нестерко. – К.: НТУУ «КПІ», 2016. – 76 с. URL : <https://ela.kpi.ua/handle/123456789/16671>.