

УХВАЛЕНО

Науково-технічною радою ТНТУ

Протокол №6 від 20 травня 2024 р

Проректор з наукової роботи

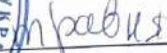
 Павло МАРУЩАК



ЗАТВЕРДЖУЮ

Головою приймальної комісії

Микола МИТНИК

 2024р

ПРОГРАМА

для вступу на навчання

для здобуття ступеня доктора філософії

за спеціальність 121 Інженерія програмного забезпечення

Схвалено вченою радою

факультет комп'ютерно-інформаційних

систем і програмної інженерії

протокол № 8 від 16 травня 2024р

Декан факультету

Ігор БАРАН 

ТЕРНОПІЛЬ 2024

ЗМІСТ

1 АНОТАЦІЯ	3
2 ЗМІСТ ПРОГРАМИ	6
2.1 АЛГОРИТМИ ТА СТРУКТУРИ ДАНИХ	6
2.2 ОБ'ЄКТНО-ОРІЄНТОВАНЕ ПРОГРАМУВАННЯ	6
2.3 АРХІТЕКТУРА І ПРОЕКТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ	8
2.4 ДИФЕРЕНЦІАЛЬНІ РІВНЯННЯ В МАТЕМАТИЧНИХ МОДЕЛЯХ	8
2.5 СИСТЕМИ РІВНЯНЬ В ЧАСТИННИХ ПОХІДНИХ	9
3 КРИТЕРІЇ ТА ШКАЛА ОЦІНЮВАННЯ ВСТУПНОГО ВИПРОБУВАННЯ	11
4 НАУКОВИЙ ТЕКСТ	12
5 ПОРЯДОК ПРОВЕДЕННЯ СПІВБЕСІДИ	13
6 КРИТЕРІЇ ОЦІНЮВАННЯ СПІВБЕСІДИ	15
7 ОСКАРЖЕННЯ РЕЗУЛЬТАТІВ ІСПИТІВ ТА СПІВБЕСІДИ	16
8 СТРУКТУРА ЕКЗАМЕНАЦІЙНОГО БІЛЕТА	17
9 РЕКЛМЕДОВАНА ЛІТЕРАТУРА	18

АНОТАЦІЯ

Метою вступних випробувань є комплексна перевірка знань вступників, які вони отримали в результаті вивчення циклу дисциплін, передбачених освітньо-професійною програмою та навчальними планами у відповідності з ступенем бакалавр та магістр за спеціальністю 121 «Інженерія програмного забезпечення».

Вступні випробування охоплюють нормативні дисципліни з циклу математичної та природничо-наукової підготовки, з також дисципліни професійної підготовки студентів відповідно до освітньо-професійної програми спеціальності 121 «Інженерія програмного забезпечення». Під час підготовки до випробування необхідно звернути увагу на те, що абітурієнт повинен:

- мати сучасні уявлення що до інженерії вимог програмного забезпечення, вміти формувати, класифікувати, аналізувати вимоги, розробляти специфікацію вимог, виконувати верифікацію та атестацію вимог;

- володіти методами моделювання програмного забезпечення та вміннями моделювати різні процеси та системи;

- володіти сучасними уявленнями про структуру та архітектуру програмного забезпечення, методи проєктування архітектури програмного Забезпечення та бути здатним проєктувати компоненти архітектури програмного продукту; моделювати процеси функціонування окремих підсистем і модулів.

- володіти навичками розробки та аналізу алгоритмів, конструювання програмного забезпечення, методами та технологіями об'єктно-орієнтованого проєктування та програмування;

- розвивати і реалізовувати нові конкурентоспроможні ідеї в інженерії програмного забезпечення;

- володіти сучасними уявленнями про інформаційні моделі та системи, реляційні та розподілені бази даних;

- вміти розв'язувати математичні, фізичні та економічні задачі шляхом створення відповідних програмних застосувань;

- володіти знаннями з математичного моделювання, диференціальних рівнянь і варіаційного числення та навичками їх застосування до розв'язання практичних завдань;

- вміти використовувати можливості апаратного забезпечення, операційних систем та мережевих технологій для створення ефективного програмного забезпечення;

- вміти забезпечувати захищеність програм та даних від несанкціонованих дій;
- розробляти, аналізувати та застосовувати специфікації, стандарти, правила і рекомендації в сфері інженерії програмного забезпечення;
- критично осмислювати проблеми у галузі інформаційних технологій та на межі галузей знань, інтегрувати відповідні знання та розв'язувати складні задачі у широких або мультидисциплінарних контекстах;
- розробляти і координувати процеси, етапи та ітерації життєвого циклу програмного забезпечення на основі застосування сучасних моделей, методів та технологій розроблення програмного забезпечення;
- забезпечувати якість програмного забезпечення.

Організація вступного випробовування здійснюється відповідно до Правил прийому Тернопільського національного технічного університету імені Івана Пулюя у 2023 році та Положення про аспірантуру ТНТУ.

Вимоги до рівня освіти осіб, які можуть розпочати навчання за освітніми програмами відповідної спеціальності, та їх результатів навчання для здобуття освітньо-наукового рівня «доктор філософії» можуть вступати особи, що здобули освітньо-професійний рівень «магістр». Програми вступних випробувань для осіб, що здобули попередній рівень вищої освіти за іншими спеціальностями, мають передбачати перевірку набуття особою таких результатів навчання:

1. Знати сучасні професійні стандарти і нормативно-правові документи з інженерії програмного забезпечення
2. Оцінювати і вибирати ефективні методи і моделі розроблення, впровадження, супроводу програмного забезпечення та управління відповідними процесами на всіх етапах життєвого циклу.
3. Розробляти, аналізувати, обґрунтовувати та систематизувати вимоги до програмного забезпечення.
4. Розробляти і оцінювати стратегії проектування програмних засобів; обґрунтовувати, аналізувати і оцінювати варіанти проектних рішень з точки зору якості кінцевого програмного продукту, ресурсних обмежень та інших факторів.
5. Застосовувати на практиці сучасні засоби розроблення програмного забезпечення.
6. Забезпечувати якість на всіх стадіях життєвого циклу програмного забезпечення, у тому числі з використанням релевантних моделей та методів оцінювання, а також засобів автоматизованого тестування і верифікації програмного забезпечення.

7. Прогнозувати розвиток програмних систем та інформаційних технологій.

8. Розробляти математичне і програмне забезпечення для наукових досліджень в галузі інженерії програмного забезпечення.

9. Планувати і виконувати наукові дослідження в сфері інженерії програмного забезпечення, обирати методики та інструменти, аналізувати результати, обґрунтовувати висновки.

Обсяг кредитів ЄКТС, необхідний для здобуття відповідного ступеня вищої освіти Освітньо-наукова програма складається з освітньої та наукової складових. Нормативний строк підготовки доктора філософії в аспірантурі становить чотири роки. Обсяг освітньої складової освітньо-наукової програми становить 51 кредит ЄКТС.

ЗМІСТ ПРОГРАМИ

ТЕМАТИЧНІ БЛОКИ

“АЛГОРИТМИ ТА СТРУКТУРИ ДАНИХ”

Поняття алгоритму. Що таке алгоритм? Властивості алгоритмів. Способи запису алгоритмів. Запис алгоритмів при допомозі блок-схем. Основні структури алгоритмів. Етапи повної побудови алгоритму. Час виконання програми. Асимптотичні співвідношення. Степінь росту функції. Визначення часу виконання алгоритму.

Абстрактні типи даних. Що таке абстрактні типи даних(АТД)? АТД “Список”. Методи реалізації “списку”: реалізація при допомозі масивів, реалізація при допомозі вказівників, реалізація при допомозі курсорів. Порівняння реалізацій. АТД “Стек”. Реалізація “стеку” при допомозі масиву. АТД “Черга” Реалізація “черги” при допомозі вказівників та при допомозі циклічного масиву. АТД “Відображення”. Реалізація “Відображення” при допомозі вказівників при допомозі списків. АТД множина. Операції що виконуються над множинами. Реалізація множин при допомозі двійкових векторів та при допомозі зв’язаних списків

Структури даних. Хеш-таблиці. Метод розв’язання колізій при допомозі ланцюжків. Закрите хешування. Реалізації хеш-функцій при закритому хешуванні. Дерева. Основна термінологія. Порядок вузлів та обхід дерева. Методи реалізації АТД “Дерева”: представлення при допомозі масивів, представлення при допомозі списків синів. Бінарні дерева. Дерево Хаффмана. Графи. Представлення орієнтованих графів. Оператори для орієнтованого графа. Задача знаходження найкоротшого шляху. Задача знаходження найкоротших шляхів між двома парами вершин. Задача знаходження існування шляхів між вершинами. Обхід орієнтованих графів. Неорієнтовані графи. Охоплююче дерево. Алгоритм Пріма.

Обчислювальні алгоритми та стратегії. Задача сортування. Прості схеми сортування: бульбашкове сортування, сортування вставками, сортування через вибір. Швидке сортування. Пірамідальне сортування. Задача пошуку. Алгоритм пошук з використанням індексації по ключах. Алгоритм послідовного пошуку. Алгоритм бінарного пошуку. Методи розробки алгоритмів. Алгоритми “Розділай та володарюй”. Динамічне програмування. “Жадібні” алгоритми.

“ОБ’ЄКТНО-ОРІЄНТОВАНЕ ПРОГРАМУВАННЯ”

Поняття ООП парадигми. Суть об’єктно-орієнтованого підходу до організації програми. Етапи об’єктно-орієнтованого програмування. Основні принципи об’єктно-орієнтованого програмування. Абстракція, інкапсуляція, наслідування й

поліморфізм. Класи і об'єкти. Синтаксис оголошення класу. Дані класу як механізм реалізації стану об'єкта. Методи класу як механізм реалізації поведінки об'єкта. Специфікатори доступу для забезпечення інкапсуляції. Статичні елементи класу. Доступ до елементів класу. Вказівники на елементи класу. Конструктори й деструктор. Конструктор за замовчуванням. Конструктор з параметрами. Конструктор копіювання. Рекомендації щодо складу класу. Операторні функції приведення типів. Дружні функції. Доступ до захищених членів класу. Дружні функції і автоматичне приведення типів. Дружні класи.

Перевантаження операцій. Основи перевантаження операторів. Вказівник `this`. Перевантаження унарних і бінарних операторів. Обмеження на перевантаження операторів. Перевантаження операторів присвоєння й індексування. Перевантажені операції `new`, `delete` і `->`. Перевантаження операторів як членів класу. Статичні та віртуальні методи. Множинне перевантаження. Поліморфізм методів. Перевизначення операторів з допомогою дружніх функцій.

Успадкування. Поняття наслідування класів. Базові і похідні класи. Форми наслідування. Конструктори похідного класу. Одиначне наслідування. Ієрархія класів. Множинне наслідування. Композиція. Конструктори при множинному успадкуванні. Включення і композиція. Правила доступу для друзів класів і похідних класів. Роль наслідування при проектуванні програм. Віртуальні функції і поліморфізм. Правила опису і використання віртуальних методів. Пізні зв'язування. Абстрактні класи і чистовіртуальні функції. Віртуальні базові класи. Відмінності структур і об'єднань від класів. Виключні ситуації. Технологія обробки виключних ситуацій. Класи виключень. Синтаксис і семантика генерації обробки виключень.

Ієрархія потокових класів. Поняття потоку введення / виведення даних в C++. Класифікація потоків. Ієрархія класів потоків. Операції введення-виведення. Стани потоків. Форматування потоків. Форматування потоків введення / виведення за допомогою функцій класу `IOS`. Функції `width()`, `precision()`, `fill()`. Маніпулятори: прості і параметризовані. Маніпулятори, визначені користувачем. Форматування потоків введення / виведення за допомогою маніпуляторів. Методи обміну з потоками. Файлові потоки: стандартні, файлові і рядкові. Файли послідовного доступу з текстовою і бінарною організацією. Потоки і типи, визначені користувачем.

Шаблони (параметризовані типи). Шаблони функцій. Шаблон простої функції. Шаблони функцій з параметрами. Вимоги до фактичних параметрів шаблону. Шаблони класів. Наслідування в шаблонах класів. Створення класів об'єктів з допомогою шаблонів. Рядки. Стандартний клас `String`. Загальні відомості. Введення-виведення об'єктів класу `String`. Опрацювання рядків з допомогою об'єктів класу `String`. Конструктори і присвоєння рядків. Операції і функції для роботи з рядками.

Бібліотека STL Контейнери. Алгоритми. Ітератори. Послідовні контейнери. Вектори. Списки. Ітератори як інтелектуальні вказівник. Загальна схема створення

контейнерів, що містять об'єкти класів користувача з використанням шаблонних класів Vector і List. Асоційовані контейнери. Множини і мультимножини. Загальна схема створення контейнерів, що містять об'єкти класів користувача з використанням шаблонних класів Set і Multiset. Відображення та мультивідображення. Загальна схема створення контейнерів, що містять об'єкти класів користувача з використанням шаблонних класів Map і Multimap.

“АРХІТЕКТУРА І ПРОЕКТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ”

Архітектура програмного забезпечення: Чому архітектура так важлива? Цілі архітектури. Принципи проектування архітектури.

Основні принципи проектування архітектури ПЗ: Основні принципи проектування. Основні питання проектування. Визначення типу програми. Вибір стратегії розгортання. Вибір відповідних технологій. Вибір показників якості. Рішення про шляхи реалізації наскрізної функціональності

Архітектурні шаблони і стилі: Що таке архітектурний стиль? Огляд основних архітектурних стилів. Архітектура клієнт / сервер. Компонентна архітектура. Проектування на основі предметної області. Багатошарова архітектура. Архітектура, заснована на шині повідомлень. N-рівнева / 3-рівнева архітектура. Об'єктно-орієнтована архітектура. Сервісно-орієнтована архітектура

Методика побудови архітектури та дизайну: Вихідні дані та етапи проектування. Визначення цілей архітектури. Час і обсяг робіт. Ключові сценарії. Важливі з точки зору архітектури варіанти використання. Загальне уявлення додатки. Відповідні технології. Графічне представлення архітектури. Питання, що потребують особливої уваги при проектуванні. Пілотні архітектури. аналіз архітектури. Оцінки на підставі сценаріїв. Подання дизайну архітектури.

“ДИФЕРЕНЦІАЛЬНІ РІВНЯННЯ В МАТЕМАТИЧНИХ МОДЕЛЯХ”

Диференціальні рівняння першого порядку та методи їх інтегрування. Рівняння зі відокремлюваними змінними. Основні поняття та означення теорії диференціальних рівнянь. Теорема існування та єдиності розв'язку. Однорідні диференціальні рівняння Однорідні та неоднорідні диференціальні рівняння, рівняння, що зводяться до них. Лінійні диференціальні рівняння першого порядку. Рівняння в повних диференціалах. Диференціальні рівняння першого порядку, що не розв'язні відносно похідної.

Диференціальні рівняння вищих порядків. Диференціальні рівняння вищих порядків, що інтегруються в квадратурах. Лінійні диференціальні рівняння вищих порядків Лінійні крайові задачі. Загальний розв'язок лінійного однорідного рівняння вищого порядку. Лінійні однорідні та неоднорідні рівняння зі сталими коефіцієнтами.

Метод фундаментальних функцій Коші для побудови розв'язку нелінійного однорідного рівняння з сталими коефіцієнтами. Метод варіації довільної сталої побудови частинного розв'язку лінійного неоднорідного диференціального рівняння.

Системи диференціальних рівнянь та їх застосування. Лінійні диференціальні рівняння n -го порядку. Фундаментальна система розв'язків. Метод варіації довільної сталої. Системи лінійних однорідних диференціальних рівнянь зі сталими коефіцієнтами. Матричний метод інтегрування лінійних диференціальних рівнянь зі сталими коефіцієнтами. Системи лінійних однорідних диференціальних рівнянь зі сталими коефіцієнтами. Лінійні неоднорідні системи диференціальних рівнянь. Побудова загального розв'язку неоднорідної системи рівнянь. Метод фундаментальних функцій Коші. Диференціальні рівняння та математичне моделювання. Диференціальні рівняння хімічних реакцій першого і другого порядку.

“СИСТЕМИ РІВНЯНЬ В ЧАСТИННИХ ПОХІДНИХ”

Загальні підходи побудови диференціальних рівнянь в частинних похідних і постановка початково-крайових задач. Аналіз методів розв'язування рівнянь в частинних похідних та крайових задач для них. Побудова диференціального рівняння в частинних похідних процесу коливань струни. Постановка початкових і крайових умов для задач і математичних моделей, що описуються рівняннями в частинних похідних. Метод Даламбера в розв'язуванні крайової задачі для рівняння в частинних похідних, що описує коливання. Метод Фур'є для розв'язування рівняння в частинних похідних в задачі про вільні коливання.

Однорідні лінійні диференціальні рівняння в частинних похідних. Однорідні лінійні диференціальні рівняння в частинних похідних і властивості їх розв'язків. Диференціальні рівняння в частинних в задачах опису процесів вимушених коливань. Розв'язання різних типів диференціальних рівнянь в частинних похідних різними методами.

Методи інтегральних перетворень для крайових задач диференціальних рівнянь в частинних похідних. Методи інтегральних перетворень Фур'є для крайових задач диференціальних рівнянь в частинних похідних для обмежених неоднорідних середовищ. Методи інтегральних перетворень Фур'є для крайових задач диференціальних рівнянь в частинних похідних для напівобмежених середовищ. Методи інтегральних перетворень Ганкеля для крайових задач диференціальних рівнянь в частинних похідних для обмежених циліндричних середовищ. Методи інтегральних перетворень Фур'є-Бесселя для крайових задач диференціальних рівнянь в частинних похідних для обмежених циліндричних середовищ. Методика застосування інтегральних перетворень Фур'є - Бесселя для крайових задач моделювання в неоднорідних циліндричних середовищах. Метод інтегрального перетворення Лапласа розв'язування крайових задач диференціальних рівнянь в

частинних похідних для однорідних середовищ. . Метод інтегрального перетворення Лапласа розв'язування крайових задач диференціальних рівнянь в частинних похідних для неоднорідних середовищ.

Системи диференціальних рівнянь в частинних похідних в математичному моделюванні фізичних процесів і явищ. Системи диференціальних рівнянь в частинних похідних в математичному моделюванні дифузії газів в пористих середовищах. Системи диференціальних рівнянь в частинних похідних в математичному моделюванні процесів перенесення вологи в середовищі вологомістких частинок.

КРИТЕРІЙ ТА ШКАЛА ОЦІНЮВАННЯ ВСТУПНОГО ВИПРОБУВАННЯ

Вступне випробування проводиться згідно програми вступного випробування за екзаменаційними білетами.

Екзаменаційний білет містить 4 питання, з дисциплін що виносяться на вступне випробування, та дають змогу перевірити теоретичні та практичні навички вступників.

Оцінювання вступного випробування відбувається за 100-бальною шкалою університету, результатам за національною шкалою («відмінно», «добре», «задовільно», «незадовільно») та шкалою Європейської кредитної трансферно-накопичувальної системи (ЄКТС – А, В, С, D, E, FХ, F) наведена у таблиці 1.

Таблиця 1

Відповідність шкал оцінювання університету, національної шкали та шкали ЄКТС

Оцінки за				
Шкалою університету	Національною шкалою		Шкалою ЄКТС	
	Екзамен	Залік		
90 – 100	Відмінно	Зараховано	А – Відмінно (вступник дає ґрунтовні, глибокі та теоретично правильні відповіді на всі поставлені питання. Послідовно, логічно, обґрунтовано, безпомилково викладає матеріал, може наводити приклади для більшої повноти відповіді);	
82 – 89			В – Дуже добре (вище середнього рівня з кількома помилками);	
75 – 81			Добре	С – Добре (вступник володіє знанням на рівні попереднього пункту, але допускає окремі несуттєві помилки, робить помилки у формулюванні окремих понять і категорій);
67 – 74				Задовільно
60 – 66			Е – Достатньо (виконання задовольняє мінімальним критеріям);	
35 – 59	Незадовільно	Не зараховано	Ф – Незадовільно (вступник дає неправильні відповіді на два-три питання екзаменаційного білету, допускає суттєві помилки, показує поверхневі знання, непослідовно викладає матеріал, невміло робить узагальнення та висновки);	
1 – 34			FX – Незадовільно (відсутність правильних відповідей).	

Вступний іспит до аспірантури вважається складеним за умови отримання здобувачем середньої оцінки комісії не нижче оцінки - «задовільно»

НАУКОВИЙ ТЕКСТ

Науковий текст зі спеціальності 121 «Інженерія програмного забезпечення» виконується вступником у разі відсутності друкованих праць з даної спеціальності у фаховому виданні на момент вступу в аспірантуру.

Науковий текст має бути дослідницького характеру і є самостійною письмовою науковою роботою. Тема наукового тексту обирається самостійно та узгоджується з науковим керівником.

Автор наукового тексту має продемонструвати :

- чітке розуміння досліджуваної проблеми, знання дискусійних питань, пов'язаних з нею;
- вміння підбирати, систематизувати та аналізувати фактичний матеріал;
- вміння формувати обгрунтовані висновки та перспективи подальших досліджень.

Структурними елементами наукового тексту є:

титульна сторінка, зміст, вступ, основна частина, висновок, список використаних джерел (обсяг 25-35 сторінок машинописного тексту формату А 4, розмір шрифту 14, інтервал 1,5).

У науковому тексті обов'язково мають міститися:

- стисла характеристика сучасної розробки проблеми, дискусійні положення і питання, що потребують вирішення, обгрунтування актуальності досліджуваної проблеми, чітке формування цілей і задач, інформаційна база дослідження;
- розкриття сутності теми, й найважливіших питань і проблем.
- результати проведеного дослідження з визначенням, наскільки вирішені поставлені завдання і досягнута мета.

ПОРЯДОК ПРОВЕДЕННЯ СПІВБЕСІДИ

Уповноважений працівник приймальної комісії забезпечує перевірку наданого вступником наукового наукового тексту (доповіді) щодо оригінальності тексту відповідно до Положення про академічну доброчесність учасників освітнього процесу ТНТУ: <http://surl.li/eksyz>. Перевірка наукових публікацій вступника за допомогою електронних сервісів не здійснюється, що не виключає можливості виявлення ознак порушень академічної доброчесності під час співбесіди вступника із предметною комісією.

Науковий текст (наукова доповідь) та/або копії наукових публікацій вступника надаються предметній комісії, яка проводить співбесіду.

Під час проходження співбесіди вступник має право:

- вільно викласти свої думки з приводу попереднього наукового досвіду (наявності наукових публікацій, участі у наукових заходах та наукових проєктах, наукових конкурсах, наукових школах, роботи в наукових гуртках, дискусійних та дебатних клубах);

- висловлюватися з приводу бажаного напрямку наукових досліджень, наукової спеціальності, ймовірного наукового керівника, можливості участі в програмах академічної мобільності, стажуваннях, у тому числі за кордоном, тощо;

- користуватися копіями наукових публікацій для підтвердження тих чи інших висловлених положень, обчислень та цитування думок інших вчених, на висновки яких посилається автор під час проведеного дослідження. Під час проходження співбесіди вступник повинен:

- бути спроможним пояснити, аргументувати і довести наукові результати, отримані автором особисто під час попереднього наукового досвіду, викладеного у наукових публікаціях (у тому числі визначення особистого внеску в наукових публікаціях, виконаних у співавторстві), чи наукової розвідки, здійсненої у вигляді наукового наукового тексту (доповіді) (актуальність обраної тематики, ступінь її розробленості у вітчизняній і зарубіжній науці, наявність власних оригінальних висновків або додаткові аргументи щодо наявних положень чи концепцій щодо предмету дослідження тощо);

- продемонструвати бачення ходу майбутнього наукового дослідження, його проміжних та кінцевих результатів, а також можливості їхнього впровадження у різні сфери людської діяльності за спеціальністю;

- продемонструвати розуміння основ наукової роботи (основ культури мовлення, академічного письма, значення наукових досліджень та наукових праць, призначення та особливостей участі у наукових заходах тощо);

- надати власні міркування щодо категорій та правил академічної доброчесності, котрі відповідають міжнародним та національним стандартам academic integrity.

За потреби предметна комісія може задавати уточнюючі питання, зокрема:

– Що стало мотивом (поштовхом) для вирішення питання про зайняття науковою діяльністю та проходження підготовки на третьому рівні вищої освіти як дослідника?

– Які чинники зумовили вибір наукової спеціальності та наукової школи? (актуальність, новизна дослідження тощо).

– Праці яких вітчизняних та зарубіжних науковців використані під час попереднього наукового досвіду?

– Які основні методи дослідження необхідно використовувати у процесі вирішення майбутнього наукового завдання?

– Які очікуються результати та їх значення для науки та практики?

– Чим зумовлене рішення зайнятися науковою діяльністю?

– У чому Ви вбачаєте актуальність теми наукової роботи?

– Як можете оцінити рівень наукової дослідженості обраної теми, наукової проблеми дослідження?

– Які складові наукової компетентності плануєте набути під час навчання в аспірантурі?

Про хід та результати співбесіди предметна комісія складає протокол співбесіди.

КРИТЕРІЇ ОЦІНЮВАННЯ СПІВБЕСІДИ

При визначенні результатів співбесіди та для прийняття рішення про рекомендацію або не рекомендацію до подальшого проходження вступником до аспірантури вступних випробувань предметна комісія, яка проводить співбесіду, враховує:

- уміння обґрунтувати свою думку, доводити свою позицію;
- уміння виявляти теоретичні та практичні проблеми певної наукової сфери;
- здатність формулювати завдання задля поставлених цілей наукового дослідження;
- уміння аналізувати та систематизувати наукові джерела та інформацію з них, розуміння основних наукових концепцій, які існують за обраним напрямом наукового дослідження;
- вміння коректно, стисло, точно відповідати на запитання.

Рішення про те, щоб вступника НЕ рекомендувати до подальшого проходження вступних випробувань предметною комісією може бути прийняте у разі, якщо:

1. Вступник не може обґрунтувати актуальність, об'єкт, предмет та новизну проведених наукових досліджень у вигляді наукових публікацій, наукового тексту (доповіді);
2. Відповіді на уточнюючі питання не відображають розуміння основ ведення наукової роботи, більшість відповідей на запитання неточні або неправильні, вступник не володіє предметом дослідження;
3. Існують обґрунтовані сумніви щодо самостійного виконання наукового тексту (доповіді), що підтверджено низкою питань особи, яка проводить співбесіду, із фіксуванням у протоколі співбесіди;
4. Виявлено випадки чи схильність вступника до аспірантури до порушення правил та стандартів академічної доброчесності;
5. Вступник не може продемонструвати розуміння основних наукових інститутів, які існують в обраній сфері наукового дослідження.

ОСКАРЖЕННЯ РЕЗУЛЬТАТІВ ІСПИТІВ ТА СПІВБЕСІДИ

Вступник має право на подання апеляції на результат вступних іспитів чи співбесіди, яка розглядається апеляційною комісією, склад та порядок роботи якої затверджуються наказом ректора ТНТУ. Апеляції подаються вступником особисто в письмовій формі не пізніше наступного дня після оголошення результатів іспиту чи співбесіди.

СТРУКТУРА ЕКЗАМЕНАЦІЙНОГО БІЛЕТА
МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ТЕРНОПІЛЬСЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ
ІМЕНІ ІВАНА ПУЛІОЯ

Вступний іспит з фаху

На освітньо-наукову програму «Інженерія програмного забезпечення» для здобуття третього (освітньо-наукового) рівня вищої освіти за спеціальністю 121 «Інженерія програмного забезпечення» Галузь знань 12 «Інформаційні технології»

ЕКЗАМЕНАЦІЙНИЙ БІЛЕТ № __

1. Теоретичне питання.
2. Теоретичне питання
3. Теоретичне питання
4. Практичне питання.

Затверджено на засіданні науково-технічної ради

Протокол № _ від _____ 2024 року

Проректор з наукової роботи ТНТУ



Павло
МАРУЦАК

РЕКОМЕНДОВАНА ЛІТЕРАТУРА

АЛГОРИТМИ ТА СТРУКТУРИ ДАНИХ:

1. 1. Data Structures and Algorithms Alfred V. Aho , John E. Hopcroft , Jeffrey D.Ullman - Pearson, 2010. - 448p.
2. 2. Introduction to Algorithms Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein - MIT Press, 2009. - 1312p.
3. L. Rivest, Clifford Stein - MIT Press, 2009. - 1312p.
4. Knuth D. The Art of Computer Programming, Volumes 1-4a. Donald Ervin Knuth - Addison- Wesley Professional, 2011. - 3168p
5. Sedgewick R. Algorithms Robert Sedgewick , Kevin Wayne - Addison- Wesley Professional, 2011. - 976p
6. Algorithms in C++, Parts 1-4: Fundamentals, Data Structure, Sorting, Searching, Third Edition Robert Sedgewick - Addison-Wesley Professional,1998. - 738p

ОБ'ЄКТНО-ОРІЄНТОВАНЕ ПРОГРАМУВАННЯ:

1. Horton's I. Beginning Visual C++@ 2013 [Е ресурс] / Ivor Horton's - Режим доступу до ресурсу: <https://www.pdfdrive.com/ivor-hortons-beginning-visual-c-2013-d46824353.html>.
2. Horton's I. Beginning C++17 From Novice to Professional / [Е. ресурс] / Ivor Horton's - Режим доступу до ресурсу: <https://www.pdfdrive.com/beginning-c17-from-novice-to-professional-e184792973.html>.
3. Stroustrup B. Programming: Principles and Practice Using C++ [Електронний ресурс] / Stroustrup Bjarne. - 2014. - Режим доступу до ресурсу: <https://www.pdfdrive.com/programming-principles-and-practice-using-c-e18903967.html>.
4. R. Lafore. Object-oriented programming in C ++. / [Ел. ресурс] / R. Lafore.-Режим доступу до ресурсу: <https://docs.google.com/file/d/0B21Hq6и9TsUHHqS3JIUmFuamc/view?resourcekey=0-MYlet9RIiEukd6CvLEHUbw>
5. Об'єктно-орієнтоване програмування: [Підручник] / В.В. Бублик. - К.: ІТ-книга, 2015. - 624 с.: іл. [Електронний ресурс]. - Режим доступу: URL:[http://itknyga.com.ua/docs/OOP final.pdf](http://itknyga.com.ua/docs/OOP%20final.pdf)
6. Глушаков С.В., Коваль А.В., Смирнов С.В. Язык программирования C++: Учебный курс.- Харьков: Фолио; М.:ООО «Издательство АСТ», 2001.-500 С. freeCodeCamp.org Object Oriented Programming (OOP) in C++ Course [Електронний ресурс] // freeCodeCamp.org - Режим доступу до ресурсу: <https://www.youtube.com/watch?v=wNOxleZLix4>.

7. Darshan University. Concepts of OOP [Електронний ресурс] // Darshan University.
Режим доступу До ресурсу:
[https://www.youtube.com/watch?v=hyJeuFWmROc&list=PLfJ4X48yC1k97-
tmIpp0HpW-lh2dv9i2](https://www.youtube.com/watch?v=hyJeuFWmROc&list=PLfJ4X48yC1k97-tmIpp0HpW-lh2dv9i2)

АРХИТЕКТУРА І ПРОЕКТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ:

1. "Architecture: Form, Space, and Order" by Francis D.K. Ching - John Wiley & Sons, 2014. - 464p.
2. "Architectural Drawing Course: Tools and Techniques for 2D and 3D Representation" by Mo Zell - Thames & Hudson, 2017. - 240p.
3. "The Architecture Reference & Specification Book: Everything Architects Need to Know Every Day" by Julia McMorrough - Rockport Publishers, 2013. - 272p.
4. "Building Construction Illustrated" by Francis D.K. Ching - John Wiley & Sons, 2014. - 480p.
5. "BIM Handbook: A Guide to Building Information Modeling for Owners, Managers, Designers, Engineers and Contractors" by Chuck Eastman, Paul Teicholz, Rafael Sacks, and Kathleen Liston - John Wiley & Sons, 2018. - 640p.

ДИФЕРЕНЦІАЛЬНІ РІВНЯННЯ В МАТЕМАТИЧНИХ МОДЕЛЯХ:

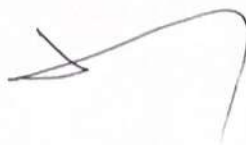
1. "Differential Equations and Their Applications: An Introduction to Applied Mathematics" by Martin Braun - Springer, 2017. - 693p.
2. "Partial Differential Equations: An Introduction" by Walter A. Strauss - John Wiley & Sons, 2008. - 464p.
3. "Applied Partial Differential Equations with Fourier Series and Boundary Value Problems" by Richard Haberman - Pearson, 2012. - 864p.
4. "Introduction to Partial Differential Equations" by Peter J. Olver - Springer, 2014. - 636p.
5. "Ordinary Differential Equations: Introduction, Fundamentals and Applications" by David Betounes - CRC Press, 2016. - 392p.
6. "Partial Differential Equations for Scientists and Engineers" by Stanley J. Farlow - Dover Publications, 2012. - 672p.
7. "Partial Differential Equations: Methods and Applications" by J. David Logan - Springer, 2014. - 430p.
8. "A Course in Ordinary Differential Equations" by Randall J. Swift, Harry F. Tiersten - CRC Press, 2017. - 630p.
9. "Partial Differential Equations for Mathematical Scientists and Engineers" by Lokenath Debnath - CRC Press, 2013. - 398p.
10. "Introduction to Partial Differential Equations with Applications" by E. C. Zachmanoglou, Dale W. Thoe - Dover Publications, 2012. - 320p.

11. "Partial Differential Equations: Analytical and Numerical Methods" by Mark S. Gockenbach - SIAM, 2018. - 728p.

СИСТЕМИ РІВНЯНЬ В ЧАСТИННИХ ПОХІДНИХ:

1. "Numerical Solution of Partial Differential Equations: Finite Difference Methods" by G. D. Smith - Oxford University Press, 2011. - 448p.
 2. "Partial Differential Equations: Methods and Applications" by J. David Logan - Springer, 2014. - 430p.
 3. "Partial Differential Equations: Modeling, Analysis, Computation" by R. M. M. Mattheij, J. H. M. ten Thije Boonkamp, B. J. A. M. Brandts - SIAM, 2018. - 506p.
 4. "Numerical Methods for Partial Differential Equations: Finite Difference and Finite Volume Methods" by Sandip Mazumder - CRC Press, 2016. - 319p.
 5. "Partial Differential Equations: Theory, Control and Approximation" by Karl-Heinz Hoffmann, Helmut Maurer - Springer, 2018. - 470p.
 6. "Introduction to Partial Differential Equations with Applications" by E. C. Zachmanoglou, Dale W. Thoe - Dover Publications, 2012. - 320p.
 7. "Partial Differential Equations in Action: Complements and Exercises" by Sandro Salsa - Springer, 2016. - 303p.
 8. "Introduction to Partial Differential Equations" by Peter J. Olver - Springer, 2014. - 636p.
 9. "Partial Differential Equations: Graduate Studies in Mathematics" by Lawrence C. Evans - American Mathematical Society, 2010. - 749p.
- "Partial Differential Equations: An Introduction with Applications" by Michael Shearer, Rachel Levy

Гарант ОНП, д.ф.-м.н. професор



Михайло ПЕТРИК