

ЗАТВЕРДЖУЮ

Голова приймальної комісії НТУ «ДП»,



ректор

О.О. Азюковський

« 15 » березня 2024 р.

ПРОГРАМА

фахового іспиту зі спеціальності

122 «Комп'ютерні науки»

для вступу на навчання за ступенем магістра

| Уміння, що контролюються | Зміст програми |
|---|---|
| <p>Виконувати об'єктно-орієнтований аналіз і моделювання реальних об'єктів процесів та явищ. Будувати ієрархічні моделі об'єктів. Створювати на базі моделей класи з використанням інструментарію об'єктно-орієнтованого програмування (інкапсуляція, наслідування, поліморфізм). Створювати шаблони проектування на базі раніше розроблених класів Використовувати концепції ООП для розробки додатків з графічним інтерфейсом користувача.</p> | <p>1 Об'єктно-орієнтоване програмування (ООП) 1.1 Моделювання реальних об'єктів процесів та явищ 1.2 Стандартні методи об'єктно-орієнтованого програмування 1.3 Шаблони функцій і класів 1.4 Використання концепцій ООП щодо розробки додатків з графічним інтерфейсом користувача</p> |
| <p>Аналізувати вимоги, розробляти специфікацію програмних засобів, виконувати їх верифікацію. Обирати оптимальні структури даних та визначати ефективні алгоритми їх обробки. Застосовувати стандарти, методи і засоби управління процесами життєвого програмного забезпечення для організації процесу його розробки відповідно до вимог і обмежень замовника. Визначати структуру програмних засобів при об'єктному підході до розробки програмного забезпечення. Будувати моделі складних систем на основі уніфікованої мови моделювання UML, застосовувати засоби моделювання для дослідження характеристик і поведінки складних об'єктів.</p> | <p>2 Технологія створення програмних продуктів 2.1 Структури даних та алгоритми їх обробки 2.2 Життєвий цикл програмного забезпечення 2.3 Об'єктний підхід до розробки програмного забезпечення 2.4 Побудова моделей складних систем на основі уніфікованої мови моделювання UML</p> |
| <p>Класифікувати задачі у залежності від критерію ефективності та архітектури операційної системи. Обирати операційну систему, що найбільш повно реалізує можливості вирішення задач користувача. Аналізувати причини переривання програм в умовах їх дослідної експлуатації. Розробляти програмні модулі з використанням сучасних технологій для синхронізації даних, управління пам'яттю, керування файлами.</p> | <p>3 Операційні системи 3.1 Архітектура операційних систем 3.2 Мультипрограмування на основі переривань 3.3 Управління пам'яттю 3.4 Файлові системи</p> |

| Уміння, що контролюються | Зміст програми |
|--|---|
| <p>Аналізувати сукупність даних задачі користувача, опираючись на сучасні теорії організації баз даних та знань.</p> <p>Обирати систему керування базами даних відповідно до поставленої задачі.</p> <p>Визначати класи фізичних об'єктів та зв'язки між ними.</p> <p>Розробляти запити на додавання, видалення або заміну даних, використовуючи мову SQL.</p> <p>Проектувати логічні та фізичні моделі баз даних.</p> | <p>4 Організація баз даних та знань</p> <p>4.1 Поняття інформаційної системи і бази даних</p> <p>4.2 Реляційна модель бази даних</p> <p>4.3 Визначення даних</p> <p>4.4 Проектування баз даних</p> |
| <p>Класифікувати архітектуру комп'ютерних мереж та функції рівнів моделі OSI.</p> <p>Аналізувати компоненти IP-адреси і функції протоколу IP.</p> <p>Розраховувати кількість вузлів в мережі і найкоротший маршрут пакету.</p> <p>Визначати типи адрес вузлів мережі, сегменти мережі для локалізації трафіка, інформаційні потоки та їх маршрути у складних мережах, засоби логічної структуризації мережі.</p> | <p>5 Комп'ютерні мережі</p> <p>5.1 Фізична структуризація мережі</p> <p>5.2 Логічна структуризація мережі</p> <p>5.3 Модель взаємодії відкритих систем OSI</p> <p>5.4 Стандартизація мереж</p> |

Рекомендована література

1. Dan Clark. Beginning C# Object-Oriented Programming (Expert's Voice in .NET) 2nd Edition, / Apress: 2018: - ISBN-13: 978-1430249351 / ISBN-10: 1430249358. 372 pp.
2. Troelsen A. Japikse P. Pro C# 9 with .NET 5: Foundational Principles and Practices in Programming: 10th edition, Apress, 2021, 1411 p.
3. Raihan Taher. Hands-On Object-Oriented Programming with C#: Build maintainable software with reusable code using C# Paperback – February 28, 2019 Publisher : Packt Publishing (February 28, 2019)/ Paperback : 288 p.- ISBN-10 : 1788296222/ ISBN-13 : 978-1788296229
4. C# Data Structures and Algorithms: Explore the possibilities of C# for developing a variety of efficient applications Paperback – Illustrated, April 19, 2018. Publisher : Packt Publishing. 292 pages. - ISBN-10 : 1788833732/ ISBN-13 : 978-178883373
5. Буров Є.В., Митник М.М. Комп'ютерні мережі : підруч. / за заг. ред. Пасічника В.В. Львів : Магнолія 2006, 2019. 334 с. ISBN: 978-617-574-121-4
6. Пасічник В.В., Резниченко В.А. Організація баз даних та знань. Київ : Видавнича група BHV, 2006. 384 с.
7. Коротєєва Т.О. Алгоритми та структури даних: навч. посіб. Львів : Видавництво Львівської політехніки. 2014. 280 с. ISBN 978-617-607-660-5
8. SWEBOOK A Complete Guide - 2021 Edition Paperback – October 15, 2020 by The Art of Service - SWEBOOK Publishing (Author) / Publisher : The Art of Service - SWEBOOK Publishing (October 15, 2020) / 314 pages / ISBN-10 : 1867419602

9. Операційні системи: навчальний посібник. / І.М. Федотова-Півень та ін.; за ред. В.М. Рудницького. Харків : ТОВ «ДІСА ПЛЮС», 2019. 216 с.
10. Stallings W. Operating Systems: Internals and Design Principles, 9th Edition. – Pearson, 2018. 800 p.
11. Постіл С.Д. UML. Уніфікована мова моделювання інформаційних систем : навч. посіб. / Ун-т держ. фіск. служби України. Ірпінь : Ун-т держ. фіск. служби України, 2019. 321 с. ISBN 978-966-337-544-1

Критерії оцінювання окремих завдань білета

Кожне теоретичне тестове завдання білета оцінюється 1 або 2 балами, а практичне та завдання на відповідність – 5 балами, виходячи з критеріїв:

а) однобальний теоретичний тест:

- 0 – вибір варіанта відповіді помилковий або обрано більш одного варіанта відповіді;
- 1 – обраний правильний варіант відповіді.

б) двобальний теоретичний тест:

- 0 – вибір варіантів відповідей помилковий або обрано більш трьох варіантів;
- 1 – лише один правильний варіант відповіді з двох обраних або два з трьох обраних;
- 2 – обрані тільки правильні варіанти відповідей.

в) практичне розрахункове завдання (задача):

- 0 – задача не вирішувалася, або були використані формули з грубими помилками, або як такі, що не належать до суті задачі;
- 1 – задача вирішувалася, але в підсумку були приведені тільки загальні формули та міркування або допущені грубі помилки у використанні формул;
- 2 – задача вирішувалася, але допущена груба помилка у формулі або в її використанні;
- 3 – задача вирішена в загальному виді, або містить грубу помилку в розрахунках, або ж відсутня пряма відповідь на запитання;
- 4 – задача вирішена в цілому правильно, але без відповідних пояснень, або допущена незначна помилка (неточність);
- 5 – задача вирішена правильно з відповідними поясненнями.

г) практичне завдання на програмування:

- 0 – програма відсутня, або у програмі використаний неправильний алгоритм, або використані дані, яких немає в умові задачі;
- 1 – у програмі конструкції мови або службові слова використано синтаксично неправильно, або програму неможливо змістовно інтерпретувати;
- 2 – у програмі деякі конструкції мови або службові слова використано синтаксично неправильно, в цілому програму можливо змістовно інтерпретувати;
- 3 – у програмі всі конструкції мови або службові слова використані синтаксично правильно, але допущені деякі помилки (відсутність необхідної кількості дужок або крапок з комою тощо);
- 4 – програма виконана повністю правильно, але відсутні коментарі;
- 5 – програма виконана повністю правильно і з відповідними коментарями.

Структура білета

Білет містить 30 однобальних теоретичних тестів, 5 двобальних, 6 п'ятибальних практичних розрахункових завдань та 6 практичних завдань на програмування, які охоплюють всі змістовні модулі програми іспиту. У підсумку максимальна сума балів білета складає 100 балів: 40 – за теоретичну частину та 60 – за практичну.

Шкала оцінювання білета

Фаховий іспит оцінюється за шкалою 100-200 балів. Мінімальний позитивний результат іспиту за виконання завдань білета (кваліфікаційний мінімум) складає 25 балів. Ця кількість балів відповідає екзаменаційній оцінці 100 шкали оцінювання. Переведення балів за виконання завдань білета вступного випробування до шкали 100-200 виконується відповідно до таблиці 5.20 додатка 5 Правил прийому до НТУ «Дніпровська політехніка». Вступники, які за результатами іспиту набрали менш ніж кваліфікаційний мінімум, позбавляються права участі в конкурсі.

Приклади екзаменаційних завдань білета

а) однобальний теоретичний тест:

Поліморфізм це:

- а) здатність функції обробляти дані різних типів;
- б) це визначення декількох методів з однаковою назвою та різними сигнатурами;
- в) механізм, завдяки якому один об'єкт може придбати властивості іншого, і додавати риси характерні тільки для нього самого;
- г) механізм, який поєднує дані та методи, що обробляють ці дані і захищає і те і інше від зовнішнього впливу або не вірного використання.

б) двобальний теоретичний тест:

З перелічених нижче процесів, вказати, які з них відносяться до основних процесів ЖЦ ПЗ:

- а) експлуатація, б) управління проектами, в) оцінка,
- г) розробка, д) створення інфраструктури проекту, е) верифікація.

в) практичне розрахункове завдання (задача):

IP-пакет розміром 4000 байт передається з в середу з MTU=1518. При передачі в середу робиться фрагментація пакету. Заголовки в цьому пакеті і в усіх його фрагментах мають однакову довжину – 18 байт. Заповніть бракуючи значення полів в заголовках фрагментів.

| | ID | Total Length | DF | MF | FragOffset | Дані (байт) |
|------------|-----|--------------|----|----|------------|-------------|
| Фрагмент 1 | | 1518 | | | | |
| Фрагмент 2 | 250 | | 0 | 1 | | 1500 |
| Фрагмент 3 | | | 0 | | | |

г) практичне завдання на програмування:

Створити програму на мові C# що, реалізує об'єктну модель яка складається з наступних класів: «рослина», «квітка», «дерево». Самостійно побудувати ієрархію класів та, у разі потреби визначити тип успадкування. Класи повинні мати необхідні поля, конструктори, в тому числі копіювання, деструктори, відповідні set і get методи. Та поліморфний метод grow, який друкує на консоль інформацію про зростання певної рослини. Програма повинна мати тестовий код, який створює масив рослин довільної кількості, що містить квіти і дерева, і, за допомогою механізму поліморфізму, виконує друк на консолі інформації про їх зростання.