

Міністерство освіти і науки України
Національний аерокосмічний університет ім. М. Є. Жуковського
«Харківський авіаційний інститут»

ЗАТВЕРДЖУЮ

Проректор з наукової роботи

В. В. Павліков

(ініціали та прізвище)

« 31 » 2020 р.

Відділ аспірантури і докторантури

**РОБОЧА ПРОГРАМА ВИБІРКОВОЇ
НАВЧАЛЬНОЇ ДИСЦИПЛІНИ**

Патерни проектування програмного забезпечення
систем критичного призначення
(назва навчальної дисципліни)

Галузь знань: 12 "Інформаційні технології"
(шифр і найменування галузі знань)

Спеціальність: 123 "Комп'ютерна інженерія"
(код та найменування спеціальності)

Освітньо-наукова програма: "Комп'ютерна інженерія"
(найменування освітньої програми)

Рівень вищої освіти: третій (освітньо-науковий)

Форма навчання: денна

Харків 2020 рік

**РОБОЧА ПРОГРАМА
ВИБІРКОВОЇ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ**

Патерни проектування програмного забезпечення систем
(назва дисципліни)
критичного призначення

для здобувачів за спеціальністю 123 "Комп'ютерна інженерія"

освітньо-наукової програми Комп'ютерна інженерія

«26» 08 2020 р., – 11 с.

Розробник: професор, к.т.н., доцент
(посада, науковий ступінь та вчене звання)


(підпис)

Орехов О.О.
(прізвище та ініціали)

Гарант ОНП доцент, к.т.н.
(посада, науковий ступінь та вчене звання)


(підпис)

Бабешко С.В.
(прізвище та ініціали)

Протокол №1 від «27» серпня 2020 р. засідання кафедри № 503

Завідувач кафедри д.т.н., професор
(науковий ступінь та вчене звання)


(підпис)

Харченко В. С.
(прізвище та ініціали)

ПОГОДЖЕНО:

Завідувач відділу

аспірантури і докторантури



В. Б. Селевко

Голова наукового товариства
студентів, аспірантів,
докторантів і молодих вчених



Т. П. Старовойт

1. Опис навчальної дисципліни

Найменування показників	Галузь знань, спеціальність, освітня програма, рівень вищої освіти	Характеристика навчальної дисципліни
		Денна форма навчання
Кількість кредитів – 5	Галузь знань 12 "Інформаційні технології" (шифр та найменування) Спеціальність 123 "Комп'ютерна інженерія" (код та найменування) Освітньо-наукова програма "Комп'ютерна інженерія" (найменування) Рівень вищої освіти: третій (освітньо-науковий)	Вибіркова
Кількість модулів – 1		Навчальний рік
Кількість змістовних модулів – 2		2020/ 2021
Індивідуальне завдання __немає (назва)		Семестр
Загальна кількість годин денна – 68/150		4-й
Кількість тижневих годин для денної форми навчання: аудиторних – 4 самостійної роботи аспіранта – 4,8		Лекції ¹⁾
		34 годин
		Практичні, семінарські ¹⁾
		0 годин
		Лабораторні ¹⁾
		34 годин
		Самостійна робота
		82 годин
		Вид контролю
іспит		

Співвідношення кількості годин аудиторних занять до самостійної роботи становить: для денної форми навчання – 68/82;

¹⁾ Аудиторне навантаження може бути зменшене або збільшене на одну годину в залежності від розкладу занять.

2. Мета та завдання навчальної дисципліни

Мета вивчення: —, полягає в оволодінні навичками написання високо ефективного коду систем критичного призначення, шляхом грамотного використання шаблонів проектування програмного забезпечення

Завдання: вміння розробляти програмне забезпечення систем критичного призначення шляхом застосування породжуючих і поведінкових патернів проектування, а також знання критеріїв за якими класифікуються патерни

Програмні компетентності. Дисципліна має допомогти сформувати у аспірантів такі компетентності:

- здатність до абстрактного мислення, аналізу та синтезу;
- здатність до пошуку, оброблення та аналізу інформації з різних джерел;
- здатність розробляти проекти та управляти ними;
- здатність застосовувати сучасні інформаційні технології, бази даних та інші електронні ресурси, спеціалізоване програмне забезпечення у науковій та навчальній діяльності;
- здатність ініціювати, розробляти і реалізовувати комплексні інноваційні проекти в комп'ютерній інженерії та дотичні до неї міждисциплінарні проекти, лідерство під час їх реалізації;
- системний науковий світогляд та загальнокультурний кругозір;
- здатність до продукування нових ідей і розв'язання комплексних проблем у галузі інформаційних технологій, а також до застосування сучасних методологій, методів та інструментів педагогічної та наукової діяльності в комп'ютерній інженерії.

Програмні результати навчання. В результаті вивчення дисципліни аспіранти мають досягти такі програмні результати навчання:

- застосовувати сучасні інструменти і технології пошуку, оброблення та аналізу інформації, зокрема, статистичні методи аналізу даних великого обсягу та/або складної структури, спеціалізовані бази даних та інформаційні системи;
- знати сучасні підходи та засоби моделювання досліджуваних об'єктів та процесів управління, в тому числі в аерокосмічній галузі, вміти створювати нові, вдосконалювати та розвивати методи математичного і комп'ютерного моделювання складних систем, оптимізації та прийняття рішень;
- знати, розуміти та вміти застосовувати методи та засоби створення інформаційних технологій та програмного забезпечення розподілених систем, Інтернету речей, хмарних обчислень, систем штучного інтелекту, віртуальної реальності у різних предметних областях, в тому числі в аерокосмічній галузі.

Міждисциплінарні зв'язки. Дисципліна є вибірковою компонентою освітньо-наукової програми «Комп'ютерна інженерія» і базується на знаннях, отриманих під час вивчення дисципліни, «ІТ в практиці наукових досліджень», «Теорія і методи зеленої ІТ-інженерії», «Наукові англомовні комунікації», «Теорія і технології критичного комп'ютингу», що є обов'язковими компонентами.

Матеріал, засвоєний під час вивчення цієї дисципліни, є базою для підготовки дисертаційної роботи.

3. Програма навчальної дисципліни

Модуль 1.

Змістовний модуль 1. Каталог патернів проєктування

Тема 1. Вступ до навчальної дисципліни.

Предмет, мета вивчення і задачі дисципліни. Структура та зміст дисципліни і методичні рекомендації щодо її вивчення. Місце дисципліни у навчальному процесі (зв'язок даного курсу з іншими дисциплінами). Вимоги до знань та вмінь тих, хто навчається. Характеристика рекомендованих під час вивчення дисципліни джерел інформації. Стислий екскурс в історію розробки програмного забезпечення.

Тема 2. Поняття патерн проєктування

Визначення. Метафора. Формат опису патернів проєктування. Назва. Класифікація. Призначення. Структура патерну на мові UML. Учасники. Відносини між учасниками. Мотивація. Застосування патерну. Результати. Реалізація.

Тема 3. Архітектурні системні патерни.

Архітектура програми. Цілі вибору архітектур. Типи архітектур і їх моделі. Класифікація архітектур. Архітектури, засновані на потоках даних. Репозиторні архітектури. Рівневі архітектури. Процедура вибору архітектури.

Тема 4. Каталог патернів проєктування

Породжуючі. Abstract Factory (Абстрактна Фабрика). Builder (Будівельник). Factory Method (Фабричний Метод). Prototype (Прототип). Singleton (Одинак). Структурні. Adapter (Адаптер). Bridge (Міст). Composite (Компоновщик). Decorator (Декоратор). Facade (Фасад). Flyweight (Пристосованець). Proxy (Заступник). Поведінкові. Chain of Responsibility (Ланцюжок Обов'язків). Command (Команда). Interpreter (Інтерпретатор). Iterator (Ітератор). Mediator (Посередник). Memento (Хранитель). Observer (Спостерігач). State (Стан). Strategy (Стратегія). Template Method (Шаблонний Метод). Visitor (Відвідувач)

Тема 5. Техніки об'єктно-орієнтованого програмування

Три найпростіші об'єктно-орієнтовані техніки, які лягли в основу категоризації каталогу патернів. Фабрика - Продукт. Фасад - Підсистема. Диспетчеризація.

Тема 6. Принципи організації каталогу

Критерії, за якими класифікуються патерни. Мета патерну. Рівень патерну. Рекомендації з вивчення патернів. Рекомендації щодо застосування патернів.

Модульний контроль.

Змістовний модуль 2. Патерни проєктування

Тема 7. Породжуючий патерн Abstract Factory

Патерн Abstract Factory. Назва. Класифікація. Частота використання. Призначення. Структура патерну на мові UML. Учасники. Відносини між учасниками. Мотивація. Застосування патерну. Результати. Реалізація. Приклад.

Тема 8. Породжуючий патерн Builder

Назва. Класифікація. Частота використання. Призначення. Структура патерну на мові UML. Учасники. Відносини між учасниками. Мотивація. Застосування патерну. Результати. Реалізація. Приклад.

Тема 9. Породжуючий патерн Factory Method

Назва. Класифікація. Частота використання. Призначення. Структура патерну на мові UML. Учасники. Відносини між учасниками. Мотивація. Застосування патерну. Результати. Реалізація. Приклад.

Тема 10. Структурний патерн Adapter

Назва. Класифікація. Частота використання. Призначення. Структура патерну на мові UML. Учасники. Відносини між учасниками. Мотивація. Застосування патерну. Результати. Реалізація. Приклад.

Тема 11. Структурний патерн Bridge

Назва. Класифікація. Частота використання. Призначення. Структура патерну на мові UML. Учасники. Відносини між учасниками. Мотивація. Застосування патерну. Результати. Реалізація. Приклад.

Тема 12. Структурний патерн Composite

Назва. Класифікація. Частота використання. Призначення. Структура патерну на мові UML. Учасники. Відносини між учасниками. Мотивація. Застосування патерну. Результати. Реалізація. Приклад.

Тема 13. Патерн поведінки Chain of Responsibility

Назва. Класифікація. Частота використання. Призначення. Структура патерну на мові UML. Учасники. Відносини між учасниками. Мотивація. Застосування патерну. Результати. Реалізація. Приклад.

Тема 14. Патерн поведінки Command

Назва. Класифікація. Частота використання. Призначення. Структура патерну на мові UML. Учасники. Відносини між учасниками. Мотивація. Застосування патерну. Результати. Реалізація. Приклад.

Тема 15. Патерн поведінки Interpreter

Назва. Класифікація. Частота використання. Призначення. Структура патерну на мові UML. Учасники. Відносини між учасниками. Мотивація. Застосування патерну. Результати. Реалізація. Приклад.

Тема 16. Патерн Iterator

Назва. Класифікація. Частота використання. Призначення. Структура патерну на мові UML. Учасники. Відносини між учасниками. Мотивація. Застосування патерну. Результати. Реалізація. Приклад.

Тема 17. Патерн Mediator

Назва. Класифікація. Частота використання. Призначення. Структура патерну на мові UML. Учасники. Відносини між учасниками. Мотивація. Застосування патерну. Результати. Реалізація. Приклад.

Модульний контроль.

4. Структура навчальної дисципліни

Назви змістовних модулів і тем	Кількість годин				
	Денна форма				
	Усього	У тому числі			
		л	п	лаб.	с. р.
1	2	3	4	5	6
Модуль 1					
Змістовний модуль 1. Каталог патернів проєктування					
Тема 1. Вступ до навчальної дисципліни	8	2			6
Тема 2. Поняття патерн проєктування	8	2			6
Тема 3. Архітектурні системні патерни	8	2			6
Тема 4. Каталог патернів проєктування	8	2			6
Тема 5. Техніки об'єктно-орієнтованого програмування	8	2			6
Тема 6. Принципи організації каталогу	9	2		1	6
Модульний контроль.	1			1	
Разом за змістовним модулем 1	50	12		2	36
Змістовний модуль 2. Патерни проєктування					
Тема 1. Породжуючий патерн Abstract Factory	12	2		4	6
Тема 2. Породжуючий патерн Builder	10	2		4	4
Тема 3. Породжуючий патерн Factory Method	10	2		4	4
Тема 4. Структурний патерн Adapter	10	2		4	4
Тема 5. Структурний патерн Bridge	10	2		4	4
Тема 6. Структурний патерн Composite	6	2			4
Тема 7. Патерн поведінки Chain of Responsibility	10	2		4	4
Тема 8. Патерн поведінки Command	10	2		4	4
Тема 9. Патерн поведінки Interpreter	6	2			4
Тема 10. Патерн Iterator	9	2		3	4
Тема 11. Патерн Mediator	6	2			4
Модульний контроль	1			1	
Разом за змістовним модулем 2	100	22		32	46
Усього годин	150	34		34	82

5. Теми семінарських занять

№ з/п	Назва теми	Кількість годин
1	<i>Не передбачено</i>	
	Разом	

6. Теми практичних занять

№ з/п	Назва теми	Кількість годин
1	<i>Не передбачено</i>	
	Разом	

7. Теми лабораторних занять

№ з/п	Назва теми	Кількість годин
1	Практична програма курсу	2
2	Породжуючий патерн Abstract Factory	4
3	Породжуючий патерн Builder	4
4	Породжуючий патерн Factory Method	4
5	Структурний патерн Adapter	4
6	Структурний патерн Bridge	4
7	Патерн поведінки Chain of Responsibility	4
8	Патерн поведінки Command	4
9	Патерн Iterator	4
	Разом	34

8. Самостійна робота

№ з/п	Назва теми	Кількість годин
1	Породжуючий патерн Prototype	6
2	Породжуючий патерн Singleton	6
3	Структурний патерн Composite	6
4	Структурний патерн Decorator	6
5	Структурний патерн Facade	6
6	Структурний патерн Flyweight	6
7	Структурний патерн Proxy	6
8	Патерн поведінки Interpreter	6

9	Патерн поведінки Iterator	6
10	Патерн поведінки Mediator	6
11	Патерн поведінки Memento	6
12	Патерн поведінки State	4
13	Патерн поведінки Strategy	4
14	Патерн поведінки Template Method	4
15	Патерн поведінки Visitor	4
	Разом	82

9. Індивідуальні завдання

Не передбачено

10. Методи навчання

Проведення аудиторних лекцій, лабораторних занять, консультацій, а також самостійна робота аспірантів за матеріалами, опублікованими кафедрою.

11. Методи контролю

Проведення поточного контролю, письмового модульного контролю, підсумковий контроль у вигляді іспиту.

12. Критерії оцінювання та розподіл балів, які отримують аспіранти

12.1. Розподіл балів, які отримують аспіранти (кількісні критерії оцінювання)

Складові навчальної роботи	Бали за одне заняття (завдання)	Кількість занять (завдань)	Сумарна кількість балів
Змістовний модуль 1			
Робота на лекціях	0...2	6	0...12
Виконання лабораторних робіт	0...5	4	0...20
Модульний контроль	0...10	1	0...10
Змістовний модуль 2			
Робота на лекціях	0...2	11	0...22
Виконання лабораторних робіт	0...5	5	0...25
Модульний контроль	0...11	1	0...11
Усього за семестр			0...100

Семестровий контроль у вигляді іспиту проводиться у разі відмови аспіранта від балів підсумкового модульного контролю й за наявності допуску

до іспиту. Під час складання семестрового іспиту аспірант має можливість отримати максимум 100 балів.

Білет для іспиту складається з одного теоретичного та одного практичного запитань, максимальна кількість за кожне із запитань, складає 50 балів.

12.2. Якісні критерії оцінювання

Необхідний обсяг знань для одержання позитивної оцінки:

- знати поняття патерн проектування;
- знати основні архітектурні системні патерни;
- знати основні техніки об'єктно-орієнтованого програмування;
- знати основні принципи організації каталогу патернів.

Необхідний обсяг вмінь для одержання позитивної оцінки:

- уміти здійснювати програмування з використанням патернів проектування;
- уміти працювати з архітектурними системними патернами;
- уміти здійснювати програмування з об'єктно-орієнтованими техніками.

12.3 Критерії оцінювання роботи аспіранта протягом семестру

Задовільно (60-74). Показати знання та уміння, виконати 90% лабораторних завдань. Знати базові поняття, що стосуються патернів проектування

Добре (75-89). Твердо знати теоретичний матеріал, опанувати матеріали всіх лекцій і виконати не менше 90% лабораторних завдань. Показати вміння виконувати всі практичні завдання в обумовлений викладачем строк з обґрунтуванням рішень та заходів, які наведено у методичних посібниках з практичних занять.

Відмінно (90-100). Повно знати основний та додатковий матеріал. Знати усі теми. Орієнтуватися у підручниках та посібниках.

Шкала оцінювання: бальна і традиційна

Сума балів	Оцінка за традиційною шкалою	
	Іспит, диференційований залік	Залік
90 – 100	Відмінно	Зараховано
75 – 89	Добре	
60 – 74	Задовільно	
0 – 59	Незадовільно	Не зараховано

13. Методичне забезпечення

1. Методи системного аналізу в радіоелектроніці та комп'ютерній інженерії: підручник / А.В. Горбенко, С.Ю. Даншина, В.А. Краснобаєв, О.О. Орехов и др.; за ред. С.Ю. Даншиної, В.С. Харченка. – Х.: Нац. аерокосм. ун-т «Харьк. авіац. ін-т», 2014. – 424 с.
2. Орехов О.О., Харченко Л.Д., Лямець В.І. Проектування програмних систем за допомогою CASE-технологій // Навч. посібник - Харків: Нац. аерокосм. ун-т «Харк. авіац. ін-т», 2005. - 83с.

14. Рекомендована література

Базова

1. Шевчук А., Охріменко Д., Касьянов А Design Patterns via C # Прийоми об'єктно-орієнтованого проектування, 2015. - 288 с.
2. Брауде Е. Технологія розробки програмного забезпечення: СПб: Пітер, 2004. - 655с.
3. Шафер Д.Ф., Фатрелл Р.Т., Шафер Л.І. Управління програмними проектами: досягнення оптимальної якості при мінімумі витрат. - М: Видавничий дім Вільямс. 2004. - 1136 с.
4. Буч Г. Об'єктно-орієнтований аналіз та проектування з прикладами додатків на C ++: СПб: Невський Діалект. 2000. - 256 с.

Допоміжна

1. Уенді Боггс, Майкл Боггс UML і Rational Rose. -М.: ЛОРИ, 2000.
2. Леонов А.В. Самовчитель UML. - СПб.: БХВ-Петербург, 2001. - 304с.
3. Трофімов С.А. Case- технології: практична робота в Rational Rose -М.: ЗАТ Видавництво БІНОМ, 2001. - 272 с.

15. Інформаційні ресурси

1. Технології проектування програмних систем. Навчальний посібник. - Чернігівський державний технологічний університет
<https://issuu.com/sveka/docs/monogr7>
2. Технології проектування програмного забезпечення
<http://www.rsatu.ru/sites/mpoevs/uploads/materials/trpo/Tehnologii%20proektirovaniya%20programmogo%20obespecheniya.218.pdf>