

Міністерство освіти і науки України

Харківський національний університет імені В.Н. Каразіна

Кафедра вищої математики та інформатики

“ЗАТВЕРДЖУЮ”
Декан факультету математики і інформатики
Григорій ЖОЛТКЕВИЧ
“ 28 ” Серпня 2024 р.

РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Шаблони об’єктно-орієнтованого програмування

(назва навчальної дисципліни)

рівень вищої освіти _____ перший(бакалаврський) _____
галузь знань _____ 01 Освіта / Педагогіка _____
(шифр і назва)
напрямок _____ 014.04 - Середня освіта(Математика) _____
(шифр і назва)
освітня програма _____ математика та інформатика _____
(шифр і назва)
спеціалізація _____
(шифр і назва)
вид дисципліни _____ за вибором _____
(обов’язкова / за вибором)
факультет _____ математики і інформатики _____

2024 / 2025 навчальний рік

Програму рекомендовано до затвердження Вченою радою факультету математики і інформатики

“ 27 ” серпня 2024 року, протокол № 8

Розробники програми:


Ігнатович Світлана Юріївна, доктор фізико-математичних наук, доцент, професор кафедри прикладної математики.

Перепелиця Олена Миколаївна, старший викладач закладу вищої освіти кафедри вищої математики та інформатики факультету математики і інформатики.

Програму схвалено на засіданні кафедри вищої математики та інформатики


Протокол від “ 27 ” серпня 2024 року № 1

Завідувач кафедри вищої математики та інформатики


_____ Віктор ЛИСИЦЯ
(підпис) (прізвище та ініціали)

Програму погоджено з гарантом освітньої - професійної програми Математика та інформатика
назва освітньої програми

Гарант освітньої - професійної програми



_____ Ганна ЧЕРНОВА
(підпис) (прізвище та ініціали)

Програму погоджено науково - методичною комісією факультету математики і інформатики

назва факультету, для здобувачів вищої освіти якого викладається навчальна дисципліна

Протокол від “ 27 ” серпня 2024 року № 1

Голова науково-методичної комісії факультету математики і інформатики


_____ Євген МЕНЯЙЛОВ
(підпис) (прізвище та ініціали)

ВСТУП

Програма навчальної дисципліни «Шаблони об'єктно-орієнтованого програмування» складена відповідно до освітньо-професійної програми «Математика та інформатика» підготовки бакалаврського рівня вищої освіти _____
(назва рівня вищої освіти)
спеціальності (предметна спеціальність) 014.04 Середня освіта (Математика)
спеціалізації _____

1. Опис навчальної дисципліни

1.1. Мета викладання навчальної дисципліни «Шаблони в об'єктно-орієнтованому програмуванні» – формування теоретичних знань та практичних навичок з розуміння та застосування основ шаблонного проектування, ознайомлення з основами узагальненого програмування.

1.2. Основні завдання вивчення дисципліни «Шаблони в об'єктно-орієнтованому програмуванні»: навчити студентів оперувати теоретичними аспектами технологій шаблонного проектування; володіти практичними вміннями і навичками розробки додатків з використанням патернів; створювати шаблони функцій і класів та застосовувати їх.

1.3. Кількість кредитів **4**

1.4. Загальна кількість годин **120**

1.5. Характеристика навчальної дисципліни	
За вибором	
Денна форма навчання	Заочна (дистанційна) форма навчання
Рік підготовки	
3-й	-й
Семестр	
5-й	-й
Лекції	
32 год.	год.
Практичні, семінарські заняття	
32 год.	год.
Лабораторні заняття	
год.	год.
Самостійна робота, у тому числі	
56 год.	год.
Індивідуальні завдання	

1.6. Заплановані результати навчання. Завершуючи навчання, студенти повинні

знати: основні принципи проектування додатків з використанням C++; призначення, класифікація та огляд патернів; структурні патерни; патерни поведінки; породжувальні патерни; патерни стратегії; приклади застосування патернів проектування додатків.

вміти: використовувати патерни проектування для розв'язання прикладних задач; створювати і використовувати шаблони функцій і класів, використовувати типові уніфіковані рішення, запобігати появленню помилок, застосовувати механізм обробки виключних ситуацій, встановлювати зв'язок програми з файлами для обміну інформацією, використовувати методи класів потоків введення/виведення.

2. Тематичний план навчальної дисципліни

Розділ 1. Об'єктно-орієнтоване програмування.

Тема 1. Основні терміни об'єктного програмування.

Зміст: Об'єкти, класи. Ієрархії класів. Наріжні камені ООП – абстракція, інкапсуляція, спадкування, поліморфізм, зв'язки між об'єктами.

Розділ 2. Принципи проектування.

Тема 1. Вступ до патернів.

Зміст: поняття патернів. Історія патернів. Користь патернів. Критика патернів. Класифікація патернів.

Тема 2. Принципи проектування.

Зміст: Якості хорошої архітектури. Повторне використання програмної архітектури та коду. Базові принципи проектування. Інкапсуляція на рівні класу, інкапсуляція на рівні методу. Програмування на рівні інтерфейсу.

Тема 3. Принципи SOLID.

Зміст: Принцип єдиного обов'язку (Single Responsibility Principle). Принцип відкритості/закритості (Open/Closed Principle). Принцип підстановки Лісков (Liskov Substitution Principle). Принцип поділу інтерфейсу (Interface Segregation Principle). Принцип інверсії залежностей (Dependency Inversion Principle).

Розділ 3. Каталог патернів. Породжувальні патерни проектування.

Тема 1. Породжувальні патерни проектування. Фабричний метод (Factory Method).

Зміст: Проблема. Рішення. Структура. Псевдокод. Застосування. Кроки реалізації. Переваги та недоліки. Відносини з іншими патернами.

Тема 2. Породжувальні патерни проектування. Абстрактна фабрика (Abstract Factory). Будівельник (Builder).

Зміст: Проблема. Рішення. Структура. Псевдокод. Застосування. Кроки реалізації. Переваги та недоліки. Відносини з іншими патернами.

Тема 3. Породжувальні патерни проектування. Прототип (Prototype). Одинак (Singleton).

Зміст: Проблема. Рішення. Структура. Псевдокод. Застосування. Кроки реалізації. Переваги та недоліки. Відносини з іншими патернами.

Розділ 4. Каталог патернів. Структурні патерни проєктування.

Тема 1. Структурні патерни проєктування. Адаптер (Wrapper, Обгортка, Adapter). Міст (Bridge).

Зміст: Проблема. Рішення. Структура. Псевдокод. Застосування. Кроки реалізації. Переваги та недоліки. Відносини з іншими патернами.

Тема 2. Структурні патерни проєктування. Компонувальник (Дерево, Composite). Декоратор (Wrapper, Обгортка, Decorator).

Зміст: Проблема. Рішення. Структура. Псевдокод. Застосування. Кроки реалізації. Переваги та недоліки. Відносини з іншими патернами.

Тема 3. Структурні патерни проєктування. Фасад (Facade). Легковаговик (Пристосуванець, Кеш, Flyweight).

Зміст: Проблема. Рішення. Структура. Псевдокод. Застосування. Кроки реалізації. Переваги та недоліки. Відносини з іншими патернами.

Тема 4. Структурні патерни проєктування. Замісник (Proxy).

Зміст: Проблема. Рішення. Структура. Псевдокод. Застосування. Кроки реалізації. Переваги та недоліки. Відносини з іншими патернами.

Розділ 5. Каталог патернів. Поведінкові патерни проєктування.

Тема 1. Поведінкові патерни проєктування. Ланцюжок обов'язків (Ланцюг відповідальностей, CoR, Chain of Command, Chain of Responsibility).

Зміст: Проблема. Рішення. Структура. Псевдокод. Застосування. Кроки реалізації. Переваги та недоліки. Відносини з іншими патернами.

Тема 2. Поведінкові патерни проєктування. Команда (Дія, Транзакція, Action, Command). Ітератор (Iterator).

Зміст: Проблема. Рішення. Структура. Псевдокод. Застосування. Кроки реалізації. Переваги та недоліки. Відносини з іншими патернами.

Тема 3. Поведінкові патерни проєктування. Посередник (Intermediary, Controller, Mediator). Знімок (Memento).

Зміст: Проблема. Рішення. Структура. Псевдокод. Застосування. Кроки реалізації. Переваги та недоліки. Відносини з іншими патернами.

Тема 4. Поведінкові патерни проєктування. Спостерігач (Видавець-Підписник, Слухач, Observer). Стан (State).

Зміст: Проблема. Рішення. Структура. Псевдокод. Застосування. Кроки реалізації. Переваги та недоліки. Відносини з іншими патернами.

Тема 5. Поведінкові патерни проєктування. Стратегія (Strategy). Шаблонний метод (Template Method).

Зміст: Проблема. Рішення. Структура. Псевдокод. Застосування. Кроки реалізації. Переваги та недоліки. Відносини з іншими патернами.

Тема 6. Поведінкові патерни проектування. Відвідувач (Visitor).

Зміст: Проблема. Рішення. Структура. Псевдокод. Застосування. Кроки реалізації. Переваги та недоліки. Відносини з іншими патернами.

3. Структура навчальної дисципліни

Назви розділів і тем	Кількість годин											
	денна форма						заочна форма					
	усього	у тому числі					усього	у тому числі				
		л	п	лаб.	інд.	с. р.		л	п	лаб.	інд.	с. р.
1	2	3	4	5	6	7	8	9	10	11	12	13
<i>Розділ 1. Об'єктно-орієнтоване програмування.</i>												
Тема 1. Основні терміни об'єктного програмування.	6	2				4						
Разом за розділом 1	6	2				4						
<i>Розділ 2. Принципи проєктування.</i>												
Тема 1. Вступ до патернів.	6	1	2			3						
Тема 2. Принципи проєктування.	5	1	2			2						
Тема 3. Принципи SOLID.	6	2	2			4						
Разом за розділом 2	17	4	6			9						
<i>Розділ 3. Каталог патернів. Породжувальні патерни проєктування.</i>												
Тема 1. Фабричний метод (Factory Method).	7	2	2			3						
Тема 2. Абстрактна фабрика (Abstract Factory). Будівельник (Builder).	8	2	2			4						
Тема 3. Прототип (Prototype). Одинак (Singleton).	8	2	2			4						
Разом за розділом 3	23	6	6			11						
<i>Розділ 4. Каталог патернів. Структурні патерни проєктування.</i>												
Тема 1. Адаптер (Wrapper, Обгортка, Adapter). Міст (Bridge).	8	2	2			3						
Тема 2. Компонувальник (Дерево, Composite).	8	2	2			3						

Декоратор (Wrapper, Обгортка, Decorator).													
Тема 3. Фасад (Facade). Легковаговик (Пристосуванець, Кеш, Flyweight).	6	2	2			2							
Тема 4. Замісник (Proxy)	6	2	2			2							
Разом за розділом 4	28	8	8			10							
<i>Розділ 5. Каталог патернів. Поведінкові патерни проєктування.</i>													
Тема 1. Ланцюжок обов'язків (Ланцюг відповідальностей, CoR, Chain of Command, Chain of Responsibility).	8	2	2			4							
Тема 2. Команда (Дія, Транзакція, Action, Command). Ітератор (Iterator).	8	2	2			4							
Тема 3. Посередник (Intermediary, Controller, Mediator). Знімок (Memento).	8	2	2			4							
Тема 4. Спостерігач (Видавець-Підписник, Слухач, Observer). Стан (State).	8	2	2			4							
Тема 5. Стратегія (Strategy). Шаблонний метод (Template Method).	8	2	2			4							
Тема 6. Відвідувач (Visitor).	6	2	2			2							
Разом за розділом 5	46	12	12			22							
<i>Усього годин</i>	120	32	32			56							

4. Теми семінарських (практичних, лабораторних) занять .

№ з/п	Назва теми	Кількість годин
1.	Повторне використання коду (спадкування, включення), практичне застосування різних видів його реалізації, їх особливості.	2
2.	Знайомство з середовищем С++ Builder. Створення додатка в програмі С++ Builder.	2
3.	Знайомство з компонентами палітри компонентів. Вибір компонентів для групових операцій. Встановлення властивостей компонентів.	2
4.	Приклад створення та виклику нової форми з головної форми програми С++ Builder. Розробка програми, яка обчислює об'єм призми.	2
5.	Створення текстового редактора.	2
6.	Робота з графікою з використанням класів, властивостей та функцій компонентів С++ Builder.	2
7.	Компонент TImage, компоненти вкладки Dialogs. Створення програми «Перегляд ілюстрацій»	2
8.	Компоненти вкладки Additional. Компонент Timer із вкладки System. Робота з датами та часом. Створення додатку «Цифровий годинник – календар»	2
9.	Створення програм: «Замовлення кави», «Улюблений напій»	2
10	Створення меню програми на прикладі Текстового редактору. Компоненти MainMenu, ScrollBar, GroupBox, RadioButton.	2
11	Створення програм: «Привітання», «Банер». Використання анімації в програмах. Створення іконки до проекту.	2
12	Компонент TabControl. Компонент PageControl. Використання цих компонентів. Створення проекту с використанням компонентів.	2
13	Створення форми для авторизації у додатку.	2
14	Робота з таблицями. Створення додатку для роботи з матрицями.	2
15	Робота з базами даних. Знайомство з компонентами вкладок: dbGo, Data Access, Data Controls палітри компонентів.	2
16	Створення проекту «Архітектурні пам'ятки Харкова».	2
	Разом	32

5. Завдання для самостійної робота

№ з/п	Види, зміст самостійної роботи	Кількість годин
1.	Приклади реалізації патернів Фабричний метод, Абстрактна фабрика. Фабричний метод	2
2.	Розробка шаблонів класів, застосування спеціалізацій шаблонів.	2

3.	Включення механізму обробки виключних ситуацій в функції та методи класів, різні випадки розміщення складових цього механізму та їх наслідки.	4
	Приклади реалізації патернів Builder, Prototype, Singleton.	4
4.	Організація обміну інформацією між програмою і файлами, різні режими встановлення зв'язку програми з файлом.	4
5.	Робота над домашніми завданнями протягом семестру, що відповідають темам практичних занять: написання програмного коду, його налагодження та використання.	4
6	Використання структурних патернів: Adapter, Bridge, Composite. Приклад шаблону декоратора C++	2
7	Використання структурних патернів: Decorator, Facade, Flyweight, Proxy.	2
	Використання патернів поведінки: Chain of Responsibility, Action, Iterator, Mediator, Memento, Observer, State, Strategy, Template Method, Visitor.	2
6.	Опрацювання теоретичного матеріалу з використанням конспекту лекцій та рекомендованої літератури.	15
7.	Підготовка до складання семестрового контролю	15
	Разом	56

6. Індивідуальні завдання

Не передбачено навчальним планом.

7. Методи навчання.

Використовуються такі види навчальних робіт: лекції, практичні заняття, самостійна робота. Усі види навчальних робіт можуть проводитися у аудиторному та дистанційному режимах. При проведенні лекцій використовуються словесні та візуальні технології навчання при активній участі студентів шляхом коротких опитувань рівня засвоєння учбового матеріалу. Самостійна робота передбачає поза аудиторне навчання, що включає підготовку до поточних аудиторних (дистанційних) занять, розбір лекційного матеріалу та вирішення типових задач, роботу з літературою, у тому числі з електронними виданнями.

8. Методи контролю

Відповіді студентів на практичних заняттях, звіти по самостійних роботах, перевірка домашнього завдання, поточний контроль на лекціях, виконання контрольних робіт.

9. Схема нарахування балів

Поточний контроль, самостійна робота, індивідуальні завдання																	Екзамен	Сума	
Розділ 1	Розділ 2			Розділ 3			Розділ 4				Розділ 5					Разом			
T1	T1	T2	T3	T1	T2	T3	T1	T2	T3	T4	T1	T2	T3	T4	T5	T6			
4	3	3	4	4	3	4	3	4	4	3	4	4	3	4	3	3	60	40	100

T1, T2 ... – теми розділів.

Для допуску до складання підсумкового контролю (екзамен) здобувач вищої освіти повинен набрати не менше 10 балів з навчальної дисципліни під час поточного контролю, самостійної роботи, індивідуального завдання.

Шкала оцінювання

Сума балів за всі види навчальної діяльності протягом семестру	Оцінка для чотирирівневої шкали оцінювання
90 – 100	відмінно
70-89	добре
50-69	задовільно
1-49	незадовільно

Критерії оцінювання

Критерії оцінювання навчальних досягнень з кожної теми

Оцінка в балах	Пояснення
Критерії оцінювання завдання у 4 бали	
4 бали	Бездоганно виконане завдання. Надання правильних відповідей на теоретичні питання з ілюстрацією їх прикладами, відповідей на додаткові запитання з теми, чим демонструє вільне володіння навчальним матеріалом; вміння застосовувати набуті теоретичні знання для вирішення практичних завдань, коли студент ефективно та вдало демонструє свої педагогічні здібності для вирішення педагогічних ситуацій; вільного володіння педагогічними методами, прийомами для вирішення фахових задач, майже не припускається помилок при розв'язанні практичних завдань.

3 бали	Наявності незначних помилок і неточностей у відповідях на теоретичні питання теми, які студент виправляє при відповіді на уточнюючі запитання; або при виникненні труднощів з наведенням прикладів чи при відповіді на додаткові запитання з теми; коли студент демонструє свої педагогічні здібності, у більшості випадків може застосовувати набуті теоретичні знання для вирішення практичних завдань; коли студент демонструє достатній рівень володіння педагогічними методами, прийомами для вирішення фахових задач, але при вирішенні практичних завдань припускається незначних помилок, які може виправити самостійно після зазначення них і/або мінімальних пояснень.
2 бали	Наявність суттєвих помилок та неточностей у відповідях на теоретичні питання теми, які студент виправляє при відповіді на уточнюючі запитання; коли студент демонструє свої педагогічні здібності, але у більшості випадків не в повному обсязі застосовує набуті теоретичні знання для вирішення практичних завдань; при вирішенні практичних завдань студент припускається значних помилок, але таких, що може виправити самостійно після зазначення на них і/або пояснень.
1 бал	Коли відповіді на теоретичні питання теми містять багато грубих помилок, виникають труднощі з наведенням прикладів, наданням відповідей на більшість додаткових і уточнюючих запитань з теми; коли у більшості випадків у студента виникають труднощі зі застосуванням теоретичних знань для вирішення практичних завдань та демонстрацією своїх педагогічних здібностей для вирішення практичних завдань; наявності багатьох грубих помилок при виконанні практичних завдань, дуже слабкий рівень володіння педагогічними методами та прийомами для вирішення фахових задач.
0 балів	Якщо студент не приступав до вирішення завдань чи зовсім не володіє теоретичними

Оцінка в балах	Пояснення
Критерії оцінювання завдання у 3 бали	
3 балів	Бездоганно виконане завдання. Надання правильних відповідей на теоретичні питання з ілюстрацією їх прикладами, відповідей на додаткові запитання з теми, чим демонструє вільне володіння навчальним матеріалом; вміння застосовувати набуті теоретичні знання для вирішення практичних завдань, коли студент ефективно та вдало демонструє свої педагогічні здібності для вирішення педагогічних ситуацій; вільного володіння педагогічними методами, прийомами для вирішення фахових задач, майже не припускається помилок при розв'язанні практичних завдань.
2 бал	Наявність суттєвих помилок та неточностей у відповідях на теоретичні питання теми, які студент виправляє при відповіді на уточнюючі запитання; коли студент демонструє свої педагогічні здібності, але у більшості випадків не в повному обсязі застосовує набуті теоретичні знання для вирішення практичних завдань; при вирішенні практичних завдань студент припускається значних помилок, але таких, що може виправити самостійно після зазначення на них і/або пояснень.
1 бал	Коли відповіді на теоретичні питання теми містять багато грубих помилок, виникають труднощі з наведенням прикладів, наданням відповідей на більшість додаткових і уточнюючих запитань з теми; коли у більшості випадків у студента виникають труднощі зі застосуванням теоретичних знань для вирішення практичних завдань та демонстрацією своїх педагогічних здібностей для вирішення практичних завдань; наявності

	багатьох грубих помилок при виконанні практичних завдань, дуже слабкий рівень володіння педагогічними методами та прийомами для вирішення фахових задач.
0 балів	Якщо студент не приступав до вирішення завдань чи зовсім не володіє теоретичними

Критерії оцінювання екзаменаційної роботи

Оцінка в балах	Критерії оцінки	Пояснення
36-40	90 – 100%	Теоретичний зміст курсу освоєний цілком, необхідні практичні навички роботи з освоєним матеріалом сформовані, всі навчальні завдання, які передбачені програмою навчання виконані в повному обсязі, відмінна робота без помилок або з однією незначною помилкою.
28-35	70 – 89%	Теоретичний зміст курсу освоєний цілком, практичні навички роботи з освоєним матеріалом в основному сформовані, всі навчальні завдання, які передбачені програмою навчання виконані, якість виконання жодного з них не оцінено мінімальним числом балів, деякі види завдань виконані з помилками, робота з декількома незначними помилками, або з однією – двома значними помилками.
20-27	50 – 69%	Теоретичний зміст курсу освоєний не повністю, але прогалини не носять істотного характеру, необхідні практичні навички роботи з освоєним матеріалом в основному сформовані, більшість передбачених програмою навчання навчальних завдань виконано, деякі з виконаних завдань, містять помилки, робота з трьома значними помилками.
0-19	1–49%	Теоретичний зміст курсу не освоєно, необхідні практичні навички роботи не сформовані, всі виконані навчальні завдання містять грубі помилки, додаткова самостійна робота над матеріалом курсу не приведе до значимого підвищення якості виконання навчальних завдань, робота, що потребує повної переробки.

10. Рекомендована література Основна література

1. Олександр Швець Занурення в патерни проектування. © Олександр Швець, Refactoring.Guru – 2022.
2. Еріх Гамма, Річард Хелм, Ральф Джонсон , Джон Вліссідес. Design Patterns: Elements of Reusable Object-Oriented Softwar – 1995.
3. Robert C. Martin Agile Software Development, Principles, Patterns, and Practices. 2021
4. Bjarne Stroustrup (The Creator of C++) The C++ Programming Language C++11, 4th Edition. 2016

Допоміжна література

1. Глинський Я. М., Анохін В. Є., Рязська В. А. С++ і С++ Builder. Навчальний посібник. Львів – 2006.

10. Посилання на інформаційні ресурси в Інтернеті, відео-лекції, інше методичне забезпечення

1. <https://codeguida.com/>
2. <https://refactoring.guru/>
3. <https://www.bestprog.net/uk/>
4. <https://github.com/septimomend/DesignPatterns-empl#articles>