

Міністерство освіти і науки України

Харківський національний університет імені В.Н. Каразіна

Кафедра вищої математики та інформатики

“ЗАТВЕРДЖУЮ”

Декан факультету математики і
інформатики
Григорій ЖОЛІСЕВИЧ



08 20 23 р.

РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Шаблони в об'єктно-орієнтованому програмуванні

(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 Середня освіта(Математика)

(шифр і назва)

вид дисципліни за вибором

(обов'язкова / за вибором)

факультет математики і інформатики

2023 / 2024 навчальний рік

Програму рекомендовано до затвердження вченою радою факультету математики і інформатики

“ 29 ” серпня 2023 року, протокол № 8

РОЗРОБНИКИ ПРОГРАМИ: (вказати авторів, їхні наукові ступені, вчені звання та посади)
Перепелиця Олена Миколаївна, старший викладач з во кафедри вищої математики та інформатики.

Програму схвалено на засіданні кафедри вищої математики та інформатики

Протокол від “ 29 ” серпня 2023 року № 1

Завідувач кафедри вищої математики та інформатики



Віктор ЛИСИЦЯ

(підпис)

(прізвище та ініціали)

Програму погоджено з гарантом освітньої (професійної/наукової) програми
(керівником проектної групи) Математика та інформатика
назва освітньої програми

Гарант освітньої (професійної/наукової) програми (керівник проектної групи)



(Ганна ЧЕРНОВА)

(підпис)

(прізвище та ініціали)

Програму погоджено методичною комісією факультету математики і інформатики
назва факультету, для здобувачів вищої освіти якого викладається навчальна дисципліна

Протокол від “ 29 ” серпня 2023 року № 1

Голова науково-методичної комісії факультету математики і інформатики



(Ольга АНОЩЕНКО)

(підпис)

(прізвище та ініціали)

ВСТУП

Програма навчальної дисципліни «Шаблони в об'єктно-орієнтованому програмуванні» складена відповідно до освітньо-професійної програми "Математика та інформатика" підготовки першого (бакалаврського) рівня вищої освіти спеціальності 014.04 Середня освіта (математика).

Викладання навчальної дисципліни, що передбачена освітньо-професійною програмою підготовки, в умовах воєнного стану та карантинних обмежень

Для проведення лекційних і практичних занять використовуються платформи Zoom, Google Meet, Skype або Google Classroom чи Viber (груповий відеозв'язок). Вибір платформи попередньо узгоджується викладачем зі студентами.

Для комунікації студентів з викладачем використовуються месенджери Viber (груповий відеозв'язок) або Telegram, а також телефонний мобільний зв'язок і електронна пошта.

Консультації проводяться за графіком, який розміщено на сайті факультету, за допомогою месенджерів Viber або Telegram та засобів телефонного або відеозв'язку (Skype або Zoom чи Google Meet).

Завдання для самостійної роботи розміщуються у групових чатах месенджерів Viber, Telegram чи у Google Classroom або надсилаються студентам електронною поштою.

Контрольні завдання здобувачі виконують за допомогою тестових платформ або письмово у текстових редакторах та надсилають викладачеві на електронну пошту.

Підсумковий контроль (іспит, залік) здійснюється у письмовій формі із надійною аутентифікацією за допомогою платформ Skype, Zoom або Google Meet. Результати надсилаються електронною поштою у визначений викладачем термін.

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	
Нормативна / за вибором	
Денна форма навчання	Заочна (дистанційна) форма навчання
Рік підготовки	

4-й	-й
Семестр	
7-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 С++; 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, запобігая появленню помилок, застосовувати механізм обробки виключних ситуацій, встановлювати зв'язок програми з файлами для обміну інформацією, використовувати методи класів потоків введення/виведення.

Загальні компетентності

- ЗК01. Здатність до абстрактного мислення, аналізу та синтезу, володіння культурою мислення.
- ЗК02. Здатність аналізувати світоглядні, соціально та особистісно значущі філософські проблеми.
- ЗК03. Здатність керуватись у своїй діяльності базовими культурними цінностями, сучасними принципами толерантності, діалогу та співробітництва.
- ЗК04. Здатність вчитися і оволодівати сучасними знаннями, використовувати знання про сучасну природничу картину світу в освітній та професійній діяльності, застосовувати знання у практичних ситуаціях.
- ЗК05. Здатність спілкуватися державною мовою, логічно вірно будувати усну та письмову мову.
- ЗК06. Здатність працювати в команді.
- ЗК07. Здатність використовувати основні методи, способи та засоби одержання, зберігання, переробки інформації.
- ЗК08. Здатність працювати з комп'ютером як засобом управління інформацією.
- ЗК09. Здатність спілкування іноземною мовою.
- ЗК10. Здатність приймати обґрунтовані рішення.

ЗК11. Здатність розуміти сутність і значення інформації в розвитку сучасного суспільства, дотримуватись основних вимог інформаційної безпеки.

Спеціальні (фахові, предметні) компетентності

ФК01. Здатність розробляти та реалізовувати навчальні програми базових і елективних курсів у різних освітніх установах.

ФК02. Здатність вирішувати задачі виховання та духовно-морального розвитку особистості учнів.

ФК03. Здатність застосовувати сучасні методики та технології, методи діагностування досягнень учнів для забезпечення якості навчально-виховного процесу.

ФК04. Здатність здійснювати педагогічний супровід процесів соціалізації та професійного самовизначення учнів, підготовки їх до свідомого вибору професії.

ФК05. Готовність до взаємодії з учнями, батьками, колегами, соціальними партнерами.

ФК06. Здатність організовувати співробітництво з учнями, підтримувати їх активність, ініціативність, самостійність та їх творчі здібності.

ФК09. Здатність здійснювати логічний аналіз математичних об'єктів і процедур та конкретизацію абстрактних математичних знань у процесі вивчення математики.

ФК10. Володіння культурами математичного мислення, логічною, алгоритмічною та евристичною; розуміння загальної структури математичного знання, взаємозв'язку між різними математичними дисциплінами; здатність користуватися мовою математики, коректно виражати та аргументовано обґрунтовувати наявні знання.

ФК11. Здатність будувати математичні моделі для вирішення практичних проблем; розуміння критеріїв якості математичного моделювання.

Програмні результати навчання

ПРН03. Знати основні поняття логіки, загальні принципи побудови математичних теорій, у тому числі аксіоматичний. Уміти формулювати та доводити математичні твердження, отримувати висновки, встановлювати правильність розв'язання задач та міркувань; володіти методами логічного виведення (дедуктивні, індуктивні, семантичні тощо). Володіти основами методу формалізації, навичками узагальнення навчальних дій, методами математичних міркувань.

ПРН11. Знати основні поняття і методи інформатики і програмування: системи числення, архітектура комп'ютера, типи і структури даних, алгоритми. Уміти використовувати існуючі алгоритми у професійній діяльності, взаємодіяти з різними суб'єктами мережного інформаційного освітнього середовища. Володіти основними методами відбору інформаційних ресурсів для супроводу навчального процесу, основними способами і методами одержання, зберігання, обробки інформації, навичками редагування текстів за допомогою пакетів прикладних програм.

ПРН20. Знати основні напрямки розвитку ІКТ і їх застосування в освітньому процесі, знати основні комп'ютерні математичні пакети, які застосовуються у навчальному процесі, основні інноваційні педагогічні технології. Уміти використовувати ІКТ для підготовки засобів діагностики і контролю, створювати прості сайти для обміну інформацією з учнями, розв'язувати типові задачі з використанням основних типів професійного математичного програмного забезпечення, застосовувати сучасні навчальні технології. Володіти навичками роботи зі спеціалізованими математичними комп'ютерними пакетами, навичками отримання інформації у комп'ютерних мережах, навичками створення простих тестових завдань з

використанням ІКТ, навичками впровадження інноваційних педагогічних технологій у навчальний процес.

ПРН22. Знати основи об'єктно орієнтованого програмування, склад та принцип роботи персонального комп'ютерів, роботу мережі INTERNET, основні функції ОС, принципи створення web-сторінок. Уміти працювати з документами і текстами, зберігати, копіювати інформацію, використовувати сервісні служби Internet, використовувати файлову систему та файлові розширення, створювати прості web-сторінки. Володіти основами програмування на мові HTML, найпростішими прийомами Web-дизайну, шаблонами об'єктно орієнтованого програмування.

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 проектування.

Тема 1. Структурні патерни проектування. 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 проектування. 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 проектування. 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 проектування. Замісник (Proxy).

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

Розділ 5. Каталог патернів. Поведінкові патерни проектування.

Тема 1. Поведінкові патерни проектування. 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 проектування. 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 проектування. 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 проектування. 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 проектування. 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.	2	2				2						
Разом за розділом 1	4	2				2						
<i>Розділ 2. Принципи проектування.</i>												
Тема 1. Вступ до патернів.	6	1	2			3						
Тема 2. Принципи проектування.	7	1	2			4						
Тема 3. Принципи SOLID.	6	2	0			4						
Разом за розділом 2	19	4	4			11						
<i>Розділ 3. Каталог патернів. Породжувальні патерни проектування.</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 проектування.</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 проектування.</i>													
Тема 1. Ланцюжок обов'язків (Ланцюг відповідальностей, CoR, Chain of Command, Chain of Responsibility).		2	2			4							
Тема 2. Команда (Дія, Транзакція, Action, Command). Ітератор (Iterator).		2	2			4							
Тема 3. Посередник (Intermediary, Controller, Mediator). Знімок (Memento).		2	2			4							
Тема 4. Спостерігач (Видавець-Підписник, Слухач, Observer). Стан (State).		2	2			4							
Тема 5. Стратегія (Strategy). Шаблонний метод (Template Method).		2	2			4							
Тема 6. Відвідувач (Visitor).		2				2							
Разом за розділом 5		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

Індивідуальні завдання: виконання курсової науково-дослідницької роботи.

7. Методи контролю

Контроль за відвідуванням занять, усне опитування під час лекцій та практичних занять, перевірка домашніх завдань, заключний семестровий контроль у формі екзамену.

8. Схема нарахування балів

Приклад для підсумкового семестрового контролю при проведенні семестрового екзамену або залікової роботи.

		Поточний контроль, самостійна робота, індивідуальні завдання															Разом	Екзамен	Сума	
Розділ 1	Розділ 2			Розділ 3			Розділ 4				Розділ 5									Індивідуальне. завдання
T1	T1	T2	T3	T1	T2	T3	T1	T2	T3	T4	T1	T2	T3	T4	T5	T6				
1	1	3	3	2	3	3	3	3	2	3	3	3	3	3	3	3	15	60	40	100

T1, T2 ... – теми розділів.

Для допуску до складання підсумкового контролю (екзамен) здобувач вищої освіти повинен набрати не менше 25 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	для дворівневої шкали оцінювання
90 – 100	відмінно	зараховано
70-89	добре	
50-69	задовільно	
1-49	незадовільно	не зараховано

Критерії оцінювання

Оцінка в балах	Оцінка за національною шкалою	
Оцінка	Пояснення	
90 – 100	Відмінно	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.
70 – 89	Добре	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.
50 – 69	Задовільно	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.
1–49	Незадовільно	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
--	--	--

9. 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., Ряжська В. А. C++ і C++ 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>