



Силабус освітнього компонента

Програма навчальної дисципліни

Об'єктно-орієнтоване програмування



Шифр та назва спеціальності

F4 – Системний аналіз та наука про дані

Інститут

ННІ Комп'ютерних наук та інформаційних технологій

Спеціалізація

Кафедра

Системного аналізу та інформаційно-аналітичних технологій (322)

Освітня програма

Системний аналіз і управління

Тип дисципліни

Спеціальна (фахова), Обов'язкова

Рівень освіти

Перший (бакалаврський)

Форма навчання

Денна, заочна

Семестр

3

Мова викладання

Українська

Викладачі, розробники



Любченко Наталія Юріївна

Nataliia.Liubchenko@khai.edu.ua

Кандидат технічних наук, доцент, доцент кафедри системного аналізу та інформаційно-аналітичних технологій

Досвід роботи – 22 років. Автор понад 100 наукових та навчально-методичних праць. Провідний лектор з дисциплін: «Основи програмування», «Технології та засоби великих даних», «Семантичний WEB», «Об'єктно-орієнтоване програмування», «Основи розподілених та паралельних обчислень», «Інженерія великих даних».

[Детальніше про викладача на сайті кафедри](#)

Загальна інформація

Анотація

В дисципліні розглядаються теорія і практика застосування основних принципів об'єктно-орієнтованого програмування, а також методи і засоби об'єктно-орієнтованого аналізу та проектування на мові C++ програмних систем різного призначення з урахуванням базових алгоритмічних структур.

Мета та цілі дисципліни

Дисципліна має метою надання теоретичних і практичних знань щодо сучасних наукових концепцій, понять, методів та технологій об'єктно-орієнтованого програмування, дослідження властивостей і особливостей класів та об'єктів, а також процесів створення працездатних програм

Формат занять

Лекції, лабораторні роботи, самостійна робота, консультації. Курсова робота, модульні тести. Підсумковий контроль – іспит.

Компетентності

ЗК2. Здатність застосовувати знання у практичних ситуаціях.

ЗК4. Знання та розуміння предметної області та розуміння професійної діяльності.

ЗК7. Здатність до пошуку, оброблення та аналізу інформації з різних джерел.

ЗК10. Здатність працювати автономно.

ЗК11. Здатність генерувати нові ідеї (креативність).

СК6. Здатність до комп'ютерної реалізації математичних моделей реальних систем і процесів; проектувати, застосовувати і супроводжувати програмні засоби моделювання, прийняття рішень, оптимізації, обробки інформації, інтелектуального аналізу даних.

СК7. Здатність використовувати сучасні інформаційні технології для комп'ютерної реалізації математичних моделей та прогнозування поведінки конкретних систем а саме: об'єктно-орієнтований підхід при проектуванні складних систем різної природи, прикладні математичні пакети, застосування баз даних і знань.

СК10. Здатність розробляти експериментальні та спостережувальні дослідження і аналізувати дані, отримані в них

Результати навчання

РН8. Володіти сучасними методами розробки програм і програмних комплексів та прийняття оптимальних рішень щодо складу програмного забезпечення, алгоритмів процедур і операцій.

РН9. Вміти створювати ефективні алгоритми для обчислювальних задач системного аналізу та систем підтримки прийняття рішень.

РН12. Застосовувати методи і засоби роботи з даними і знаннями, методи математичного, логіко-семантичного, об'єктного та імітаційного моделювання, технології системного і статистичного аналізу.

Обсяг дисципліни

Загальний обсяг дисципліни 150 год. (5 кредитів ECTS): лекції – 32 год., лабораторні роботи – 32 год., самостійна робота – 86 год.

Передумови вивчення дисципліни (пререквізити)

Алгоритмізація та програмування.

Особливості дисципліни, методи та технології навчання

Лекції проводяться інтерактивно з використанням мультимедійних технологій.

Навчальні матеріали, в тому числі відеозаписи лекцій, доступні студентам через засоби Microsoft Teams. Вивчення курсу потребує використання програмного забезпечення Visual Studio (або інш.), крім загально вживаних програм і операційних систем.

Програма навчальної дисципліни

Навчальні заняття

Лекції

Кількість годин

Теми лекцій

Тема 1. Загальні положення мови C++. Введення в класи Прості класи. Доступ до елементів класу.	2
Тема 2. Конструктори та деструктори Конструктори за умовчанням. Конструктори з параметрами. Конструктор копіювання. Масиви об'єктів класу. Передача об'єктів в функції. Статичні елементи класу.	2

Тема 3. Перевантаження функцій та операторів як членів класу	4
Перевантаження конструкторів та конструктори копіювання. Перевантаження операторів присвоєння та індексування. Перевантаження бінарних операторів. Перевантаження унарних операторів. Перевантаження операторів інкременту та декременту.	
Тема 4. Перевантаження функцій та операторів як дружніх функцій	2
Дружні оператор-функції. Перевантаження операторів введення/ виведення.	
Тема 5. Механізм успадкування класів	2
Ієрархія класів. Доступ до елементів. Принципи успадкування. Доступ до елементів класу. Захищені елементи класу. Конструктори та деструктори при успадкуванні.	
Тема 6. Множинне успадкування	2
Множинне успадкування. Віртуальні базові класи. Віртуальні функції. Абстрактні базові класи.	
Тема 7. Форматування стандартних потоків. Введення/виведення у C++	2
Форматне введення/виведення. Функції. Прапорці. Маніпулятори. Введення/виведення та потоки. Створення власних функцій вставки до потоку. Створення власних функцій вилучення з потоку. Створення власних маніпуляторів.	
Тема 8. Шаблонні функції та класи	2
Опис та використання шаблонних функцій. Порядок звернення до шаблонних функцій. Перевантаження шаблонної функції. Основні операції над ітераторами. Шаблони класів. Родові функції. Родові класи.	
Тема 9. Обробка виключних ситуацій	2
Класифікація виключних ситуацій. Загальний механізм обробки виключень. Використання класу як типу виключення. Шаблони і обробка виняткових ситуацій.	
Тема 10. Тестування	2
Тестування. Функція assert.	
Тема 11. Використання засобів Стандартної бібліотеки C++ (STL)	4
Стандартні контейнерні класи. Алгоритми. Класи-контейнери. Вектори. Списки. Асоціативні контейнери. Функтори.	
Тема 12. Алгоритми STL. Лямбда-вирази	4
Класифікація алгоритмів. Немодифікуючі алгоритми, модифікуючі алгоритми, алгоритми сортування, чисельні алгоритми. Лямбда-вирази. Використання лямбда-функцій в алгоритмах. Специфікатор mutable.	
Тема 13. Рядковий клас	2
Операції над рядками. Функції роботи з рядками.	
Загальна кількість годин	32

Практичні заняття

Практичні роботи в рамках дисципліни не передбачені.

Лабораторні заняття

Теми лабораторних занять	Кількість годин	Вагові коефіцієнти <i>a</i>
Тема 1. Прості класи. Доступ до елементів класу	4	2
Класи, посилання, аргументи за умовчанням. Масиви об'єктів класу. Передача об'єктів в функції.		

Тема 2. Перевантаження операторів як членів класу Перевантаження бінарних та унарних операторів.	4	2
Тема 3. Перевантаження операторів як дружні функції Перевантаження бінарних та унарних операторів.	2	1
Тема 4. Успадкування Базові класи. Доступ до елементів класу. Захищені елементи класу. Конструктори та деструктори при успадкуванні.	4	2
Тема 5. Множене успадкування Ієрархії класів. Віртуальні функції та класи.	2	1
Тема 6. Файли Використання бібліотеки введення/виводу в мові C++. Створення власних маніпуляторів.	4	2
Тема 7. Шаблонні функції та класи Створення власних шаблонів.	2	1
Тема 8. Обробка виключних ситуацій	2	1
Тема 9. Стандартна бібліотека шаблонів (STL) Вектори. Списки. Асоціативні контейнери. Функтори. Алгоритми STL. Лямбда-вирази.	6	3
Тема 10. Рядковий клас Операції над рядками. Функції роботи з рядками.	2	1
Загальна кількість годин	32	$\sum_{i=1}^n a_i = 16$

Контрольні роботи

Теми контрольних тестів	Вагові коефіцієнти b
Тест 1. Прості класи. Перевантаження операторів як членів класу. Перевантаження операторів як дружні функції. Успадкування. Множене успадкування	2
Тест 2. Файли. Шаблонні функції та класи. Обробка виключних ситуацій. Стандартна бібліотека шаблонів (STL). Рядковий клас	2
Загалом	$\sum_{i=1}^m b_i = 4$

Самостійна робота

До самостійної роботи відноситься самостійне опрацювання теоретичного матеріалу; підготовка до проходження тестів; написання тексту програм згідно завдання до лабораторних робіт та аналіз отриманих результатів; підготовка звіту до лабораторних робіт та виконання індивідуального завдання (курсової роботи).

Опрацювання теоретичного матеріалу

Теми для самостійного вивчення	Кількість годин
Тема 1. SOLID-принципи SOLID-принципи: вплив на проектування класів.	2

Тема 2. Патерн проектування	20
Розробка програм з використанням породжуючих патернів (Creational patterns). Шаблон Абстрактна фабрика (Abstract Factory), Шаблон Одинак (Singleton), Шаблон Фабричний метод (Factory Method). Розробка програм з використанням структурних патернів (Structural patterns). Шаблон Декоратор (Decorator), Шаблон Проксі (Proxy), Шаблон Мост (Bridge). Розробка програм з використанням патернів поведінки (Behavioral patterns). Шаблон Стратегія (Strategy), Шаблон Медіатор (Mediator)	
Тема 3. Відмінності між статичним і динамічним поліморфізмом	2
Тема 4. Розробка статичних та динамічних .dll бібліотек	6
Модульне тестування. Об'єктні файли. Динамічне компонування. Неявне і явне зв'язування.	
Тема 5. Кросплатформені багатопотокові бібліотеки	6
Ідентифікатор потоку. Засоби синхронізації. Події. М'ютекси і семафори. Класична задача «виробник-споживач». Класична задача про читачів-письменників.	
Загальна кількість годин	36

Тематика індивідуальних завдань

Виконання курсової роботи (КР) передбачає розробку в середовищі Visual Studio фірми Microsoft (або інше) консольного застосування для роботи з ієрархією класів, відповідно до варіанту. Обсяг звіту: 25-30 сторінок основного тексту. Завдання виконується протягом навчальних тижнів і подається на перевірку до екзамену. У процесі виконання студент має продемонструвати вміння застосовувати теоретичні знання на практиці, працювати з інструментами середовища програмування та забезпечувати коректність і зручність роботи створеного додатку.

Етапи виконання КР

1. Аналіз задачі	
Дослідження предметної області (наприклад: база даних студентів, книг, товарів тощо). Визначення функціональних вимог до програми. Вибір методів і технологій для реалізації.	
	5
2. Проектування	
Визначення структури класів та їх ієрархії. Розробка UML-діаграм (класів, варіантів використання, послідовності). Проектування графічного інтерфейсу користувача (вікно програми, меню, кнопки, таблиці).	
	10
3. Реалізація програми	
Створення класів і реалізація методів відповідно до завдання. Організація роботи головного меню та інтерфейсу користувача. Реалізація функцій введення, редагування, пошуку, видалення та перегляду даних. Додавання функцій збереження та завантаження даних із файлу.	
	20
4. Тестування	
Перевірка роботи всіх функцій програми. Тестування з коректними та некоректними даними. Аналіз отриманих результатів.	
	5
5. Оформлення звіту	
Реферативна частина (аналіз задачі, обґрунтування вибору методів). Теоретична частина (вимоги до ПЗ, UML-діаграми, проектування інтерфейсу). Програмна частина (реалізація, тестування, скріншоти). Висновки щодо виконаної роботи.	
	10
Загальна кількість годин	50

Неформальна освіта

Здобувач має можливість перезарахувати окремі теми або курс шляхом: проходження професійних курсів чи тренінгів, онлайн-освіти, професійних стажувань, у сфері, що відповідає навчальним цілям дисципліни. Для зарахування необхідно надати: сертифікат про проходження курсу/стажування, опис програми тренінгу із зазначенням змісту тем, обсягу і тривалості.

Рекомендовані курси, тренінги, стажування

1. Онлайн-курс «Introduction to Object-Oriented Programming in C++»
<https://www.coursera.org/learn/cplusplus-crypto-i?specialization=object-oriented-programming-s12n>
2. Онлайн-курс «C++ Programming: Classes and Data»
<https://www.coursera.org/learn/cplusplus-crypto-ii?specialization=object-oriented-programming-s12n>
3. Онлайн-курс «Object-Oriented Programming in C++: Functions»
<https://www.coursera.org/learn/cplusplus-crypto-iii?specialization=object-oriented-programming-s12n>

Література, навчальні матеріали та інформаційні ресурси

Основна література

1. «Об'єктно-орієнтоване програмування. Практикум»: навч. посіб. – КПІ ім. Ігоря Сікорського; уклад. В.А. Данілова.– КПІ ім. Ігоря Сікорського.- 2021. – 121 с
<https://ela.kpi.ua/items/d9760ade-916b-449f-9a7e-a8c1cf42bf14>
2. Гришанович Т. О., Глинчук Л. Я. Основи об'єктно-орієнтованого програмування : навч. посібник / ВНУ імені Лесі Українки. – Луцьк : ВНУ імені Лесі Українки, 2022. – 120 с. [Електронний ресурс]. – URL: <https://evnuir.vnu.edu.ua/bitstream/123456789/20320/1/ooop.pdf>
3. Іванов Є.О., Ліндер Я.М., Жереб К.А. Основи мови програмування C++): [Електронний ресурс] : навч. посіб., К.: Логос, 2020. – 90 с.
<https://iss.csc.knu.ua/library/study-guides/foundations-of-cplusplus-language.pdf>
4. Зеленський О.С., Лисенко В.С. Об'єктно-орієнтоване програмування на C++. Навчальний посібник. Кривий Ріг: Державний університет економіки і технологій, 2023. 215 с. [Електронний ресурс]. – Режим доступу:
https://dspace.duet.edu.ua/jspui/bitstream/123456789/827/1/%D0%9C%D0%92%20OOP_%2B%D0%92%20V.pdf
5. Порєв В.М. Об'єктно-орієнтоване програмування: конспект лекцій. – Київ : КПІ ім. Ігоря Сікорського, 2022. 271 с. [Електронний ресурс]. – Режим доступу:
https://ela.kpi.ua/bitstream/123456789/51571/1/konsp_lect_ooop_porev.pdf

Додаткова література

1. Любченко Н. Ю., Соболев М.О., Паржин Ю.В., Пугачов Р.В. Основи програмування на C/C++ в прикладах. Частина 1. / Навчально-методичний посібник. – Харків: НТУ "ХПІ". – 2021. – 120 с. [Електронний ресурс]. – URL: <http://repository.kpi.kharkov.ua/handle/KhPI-Press/54410>
2. Любченко Н. Ю., Соболев М.О., Паржин Ю.В., Пугачов Р.В., Івашко А.В. Основи програмування на C/C++ в прикладах. Частина 2. / Навчально-методичний посібник. – Харків: НТУ "ХПІ". – 2022. – 200 с. [Електронний ресурс]. – URL: <http://repository.kpi.kharkov.ua/handle/KhPI-Press/57740>
3. Уроки програмування на C++ : [Електронний ресурс]. URL: <https://acode.com.ua/urok-1-vvedennya-v-programuvannya/>
4. International Standard ISO/IEC 14882:2014(E) – Programming Language C++ , ISBN-13: 978-0321563842: [Електронний ресурс]. URL: <https://isocpp.org/std/the-standard>.
5. C++11 - the new ISO C++ standard: [Електронний ресурс]. URL: <https://www.stroustrup.com/C++11FAQ.html>
6. C library [Електронний ресурс]. – URL: <https://cplusplus.com/reference/clibrary/>
7. C++. Бібліотека стандартних шаблонів STL. Загальні поняття : [Електронний ресурс]. – URL: <https://www.bestprog.net/uk/2021/09/26/c-standard-template-library-general-concepts-ua/>.

Система оцінювання

Підсумкова оцінка з освітнього компонента визначається відповідальним лектором за темами, видами занять, тощо у відповідності до силабусу і є інтегральною оцінкою результатів усіх вид навчальної діяльності здобувача вищої освіти. Підсумкова оцінка повинна відображати всі оцінки за складовими навчального процесу з урахуванням їх вагових показників k :

Поточний контроль (практичні, семінарські, лабораторні заняття), k_1	Контрольні роботи (за наявності), k_2	Індивідуальне завдання (за наявності), k_3	Підсумковий контроль (для ОК з іспитом), k_4
0,4	0,1	0,4	0,1

Сума коефіцієнтів повинна складати одиницю: $k_1 + k_2 + k_3 + k_4 = 1$. Підбір вагових коефіцієнтів підсумкової оцінки здійснює розробник курсу.

Розрахунок підсумкової оцінки проводиться за формулою:

$$O = П \cdot k_1 + K \cdot k_2 + I \cdot k_3 + Пк \cdot k_4$$

де: $П$ – середньозважена середня оцінка за поточний контроль
 I – оцінка за виконання індивідуального завдання
 K – середньозважена оцінка за контрольні роботи
 $Пк$ – оцінка за підсумковий контроль

$$П = \frac{П_1 \cdot a_1 + П_2 \cdot a_2 + \dots + П_n \cdot a_n}{\sum_{i=1}^n a_i}$$

де: a_i - ваговий коефіцієнт за кожне практичне (семінарське) або лабораторне заняття.

$$K = \frac{K_1 \cdot b_1 + K_2 \cdot b_2 + \dots + K_m \cdot b_m}{\sum_{i=1}^m b_i}$$

де: b_i - ваговий коефіцієнт за кожну контрольну роботу.

Поточні оцінки за кожну складову ($П, K, I, \dots$) виставляються за 100-бальною шкалою згідно з [положенням «Про критерії та систему оцінювання знань та вмінь і про рейтинг здобувачів вищої освіти» НТУ «ХПІ»](#).

Підсумкова оцінка виставляється відповідно до розрахованої O з округленням до найближчого цілого числа в більшу сторону.

Шкала оцінювання

Сума балів	Національна оцінка	ECTS
90–100	Відмінно	A
82–89	Добре	B
75–81	Добре	C
64–74	Задовільно	D
60–63	Задовільно	E
35–59	Незадовільно (потрібне додаткове вивчення)	FX
1–34	Незадовільно (потрібне повторне вивчення)	F

Норми академічної етики і політика курсу

Здобувач вищої освіти повинен дотримуватися «Кодексу етики академічних взаємовідносин та доброчесності НТУ «ХПІ»: виявляти дисциплінованість, вихованість, доброзичливість, чесність, відповідальність. Конфліктні ситуації повинні відкрито обговорюватися в навчальних групах з викладачем, а при неможливості вирішення конфлікту – доводитися до відома співробітників дирекції інституту.

Нормативно-правове забезпечення впровадження принципів академічної доброчесності НТУ «ХПІ» розміщено на сайті: <http://blogs.kpi.kharkov.ua/v2/nv/akademichna-dobrochesnist/>

Погодження

Силабус погоджено

30.08.2025

Завідувач кафедри

Тетяна АЛЕКСАНДРОВА

30.08.2025

Гарант ОП

Юрій ДОРОФЄЄВ