



Силабус освітнього компонента

Програма навчальної дисципліни



Об'єктно-орієнтоване програмування та проектування

Шифр та назва спеціальності
122 – Комп'ютерні науки

Інститут
ННІ Комп'ютерного моделювання, прикладної
фізики та математики

Освітня програма
Комп'ютерні науки. Моделювання,
проектування та комп'ютерна графіка

Кафедра
Математичне моделювання та інтелектуальні
обчислення в інженерії (161)

Рівень освіти
Бакалавр

Тип дисципліни
Спеціальна (фахова), Обов'язкова

Семестр
2

Мова викладання
Українська

Викладачі, розробники



Розова Людмила Вікторівна
(відповідальний лектор)

Lyudmyla.Rozova@khpi.edu.ua

Кандидат технічних наук, доцент

Фахівець в галузі розробки наукоємного програмного забезпечення, математичного моделювання, обробки та аналізу даних, організації проектного навчання. Автор понад 40 наукових та методичних праць.

[Детальніше про викладача на сайті кафедри](#)



Дьомін Олексій Володимирович
(асистент з лабораторних робіт)

Oleksii.Domin@khpi.edu.ua

Аспірант, асистент

[Детальніше про викладача на сайті кафедри](#)



Сенько Альона Володимирівна
(асистент з лабораторних робіт)

Alyona.Senko@khpi.edu.ua

Доктор філософії

[Детальніше про викладача на сайті кафедри](#)



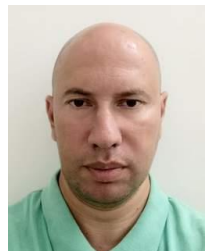
Овсяніков Владислав Валерійович

(асистент з лабораторних робіт)

Vladyslav.Ovsianikov@khpi.edu.ua

Аспірант, Senior data engineer Eram systems

[Детальніше про викладача на сайті кафедри](#)



Удалов Дмитро Володимирович

(асистент з лабораторних робіт)

Dmytro.Udalov@khpi.edu.ua

Аспірант, інженер програміст

[Детальніше про викладача на сайті кафедри](#)

Загальна інформація

Анотація

Дисципліна "Об'єктно-орієнтоване програмування та проектування" знайомить студентів з основами об'єктно-орієнтованого програмування (ООП), включаючи принципи інкапсуляції, наслідування та поліморфізму. Курс покриває методики проектування та розробки програмного забезпечення з використанням об'єктно-орієнтованих мов програмування на прикладі C++, і наголошує на розробці ефективних програмних рішень.

Мета та цілі дисципліни

Сформувані у студентів знання, вміння і навички, необхідні для усвідомлення і раціонального використання понять об'єктно-орієнтованого програмування. Задачі курсу: вивчення студентами основних принципів об'єктно-орієнтованого програмування та застосування цих принципів під час проектування та розробки програм; освоєння мови об'єктно-орієнтованого програмування C++; формування навичок декомпозиції поставленої задачі і її подальшої програмної реалізації на основі об'єктно-орієнтованих технологій.

Формат занять

Лекції, лабораторні роботи, самостійна робота, консультації. Підсумковий контроль – екзамен.

Компетентності

ЗК2: Здатність застосовувати знання у практичних ситуаціях.

ЗК9: Здатність працювати в команді.

ЗК12: Здатність оцінювати та забезпечувати якість виконуваних робіт.

СК3: Здатність до логічного мислення, побудови логічних висновків, використання формальних мов і моделей алгоритмічних обчислень, проектування, розроблення й аналізу алгоритмів, оцінювання їх ефективності та складності, розв'язності та нерозв'язності алгоритмічних проблем для адекватного моделювання предметних областей і створення програмних та інформаційних систем

СК4: Здатність використовувати сучасні методи математичного моделювання об'єктів, процесів і явищ, розробляти моделі й алгоритми чисельного розв'язування задач математичного моделювання, враховувати похибки наближеного чисельного розв'язування професійних задач.

СК8: Здатність проектувати та розробляти програмне забезпечення із застосуванням різних парадигм програмування: узагальненого, об'єктно-орієнтованого, функціонального, логічного, з відповідними моделями, методами й алгоритмами обчислень, структурами даних і механізмами управління

Результати навчання

ПР1: Застосовувати знання основних форм і законів абстрактно-логічного мислення, основ методології наукового пізнання, форм і методів вилучення, аналізу, обробки та синтезу інформації в предметній області комп'ютерних наук

ПР5: Проектувати, розробляти та аналізувати алгоритми розв'язання обчислювальних та логічних задач, оцінювати ефективність та складність алгоритмів на основі застосування формальних моделей алгоритмів та обчислюваних функцій

ПР6: Використовувати методи чисельного диференціювання та інтегрування функцій, розв'язання звичайних диференціальних та інтегральних рівнянь, особливостей чисельних методів та можливостей їх адаптації до інженерних задач, мати навички програмної реалізації чисельних методів

ПР14: Застосовувати знання методології та CASE-засобів проектування складних систем, методів структурного аналізу систем, об'єктно-орієнтованої методології проектування при розробці і дослідженні функціональних моделей організаційно-економічних і виробничо-технічних систем

Обсяг дисципліни

Загальний обсяг дисципліни 150 год. (5 кредитів ECTS): лекції – 16 год., лабораторні роботи – 48 год., самостійна робота – 86 год.

Передумови вивчення дисципліни (пререквізити)

Для успішного проходження дисципліни необхідно мати знання та практичні навички з наступних дисциплін: “Алгоритмізація та програмування”, “Вступ до спеціальності. Ознайомча практика”, “Архітектура обчислювальних систем”.

Особливості дисципліни, методи та технології навчання

Заняття з курсу “Об'єктно-орієнтоване програмування та проектування” проводяться інтерактивно, з використанням мультимедійних технологій. На лабораторних заняттях застосовується практично-орієнтований підхід до навчання, виконуються загальні та індивідуальні завдання.

Для практичного опрацювання основ об'єктно-орієнтованого програмування та виконання індивідуальних завдань лабораторних робіт використовується вільне кросплатформне середовище для розробки програм на мові C++ Code::Blocks.

Студент зобов'язаний відвідувати всі заняття згідно розкладу, виконувати лабораторні роботи. З метою оволодіння необхідною якістю освіти з дисципліни потрібно відвідуваність і регулярна підготовленість до занять.

Програма навчальної дисципліни

Теми лекційних занять

Тема 1. Введення в об'єктно-орієнтоване програмування (ООП)

Порівняння структурного та об'єктно-орієнтованого підходу у програмуванні. Основні поняття ООП. Введення у поняття класу. Поля та методи класу. Екземпляр класу - об'єкт. Основні принципи ООП. Застосування понять абстракції та інкапсуляції для створення класу. Режими доступу в класі. Поняття UML-діаграми класу. Відділення інтерфейсу класу від реалізації, багатофайлові проекти.

Тема 2. Конструктори і деструктор класу. Дружні функції і класи.

Поняття конструктора та деструктора класу. Типи конструкторів класу. Конструктор за замовчуванням. Конструктор з параметрами, у тому числі с параметрами за замовчуванням. Делегуючий конструктор. Конструктор копіювання. Показчик this. Перевантаження функцій. Дружні функції. Дружні класи.

Тема 3. Перевантаження операторів і операцій

Поняття перевантаження операторів і операцій. Правила перевантаження. Поняття операторної функції. Способи визначення операторної функції.

Перевантаження операторів з використанням методів класу. Перевантаження операторів з використанням дружніх функцій. Перевантаження операторів з використанням зовнішніх функцій

Тема 4. Статичні елементи. Успадкування

Статичні змінні та методи.

Поняття успадкування. Види успадкування. Синтаксис успадкування. Керування доступом до членів базового класу. Просте успадкування. Виклик конструкторів класу та передача параметрів конструкторам базових класів. Множинне успадкування. Проблеми множинного успадкування. Виклик деструкторів при успадкуванні. Види взаємовідношень між класами.

Тема 5. Віртуальні функції, абстрактні класи

Поняття поліморфізму. Віртуальні функції. Поліморфний клас. Правила оголошення та використання віртуальних функцій. Раннє статичне та пізнє динамічне зв'язування. Таблиця віртуальних методів. Абстрактний клас. Суто віртуальна функція. Віртуальний деструктор. Віртуальне успадкування.

Тема 6. Виключні ситуації

Типи помилок, які можуть виникати в програмах. Поняття виключної ситуації. Обробка виключної ситуації. Алгоритм обробки не перехоплених виключень. Специфікація виключень. Список виключень функції. Виключення в конструкторах і деструкторах. Класи виключень та їх ієрархія. Стандартні виключення. Клас exception.

Тема 7. Шаблони функцій і класів

Поняття шаблонів функцій та їх формат. Правила роботи зі шаблонами функцій. Шаблони класів в C++. Повна та часткова спеціалізація шаблонів класу. Статичні елементи шаблонів. Успадкування шаблонів класу

Тема 8. Простори імен. Директиви препроцесору.

Глобальний простір імен. Конфлікт імен. Оголошення просторів імен. Вкладеність просторів імен. Директиви препроцесору. Макровизначення. Умовна компіляція.

Теми лабораторних робіт

Тема 1. Повторення матеріалу першого семестру з програмування. Структури.

Лабораторна робота 1. Структури даних. Робота з бінарними файлами

Тема 2. Введення в класи і об'єкти. Застосування принципів абстракції та інкапсуляція на практичних прикладах створення класів та їх об'єктів. Створення багатофайлових проектів.

Лабораторна робота 2. Класи та об'єкти. Абстракція та інкапсуляція. Основи UML для представлення класів. Класи в C++

Тема 3. Створення різних типів конструкторів класів та деструкторів. Робота покажчика this. Створення дружніх функцій і класів

Лабораторна робота 3. Конструктори класу. Деструктор класу.

Тема 4. Перевантаження операцій і операторів трьома способами. Види операцій

Лабораторна робота 4. Перевантаження операторів.

Лабораторна робота 5. Перевантаження операторів

Тема 5. Приклади створення ієрархії класів при успадкуванні. Конструктори та деструктор при успадкуванні, особливості передачі параметрів конструкторам базових класів. Приклади та застосування статичних елементів

Лабораторна робота 6. Глобальні змінні в ООП: статичні члени класу. Шаблон проектування "Одинак".

Лабораторна робота 7. Успадкування

Тема 6. Створення віртуальних методів на прикладах. Розробка абстрактних класів. Реалізація віртуальних методів. Приклади.

Лабораторна робота 8. Поліморфізм в C++. Віртуальні функції.

Тема 7. Розглядання алгоритму обробки виключних ситуацій на прикладах. Створення класів і ієрархії класів виключень.

Лабораторна робота 9. Виключні ситуації.

Тема 8. Приклади створення шаблонів функцій і класів.

Лабораторна робота 10. Шаблони класів.

Самостійна робота складається з наступних компонентів

Самостійна робота передбачає опрацювання лекційного матеріалу, підготовка до лабораторних занять, виконання індивідуальних розрахункових завдань до кожної лабораторної роботи.

Додаткові завдання для самостійної роботи за темами лекцій та лабораторних робіт:
<http://repository.kpi.kharkov.ua/handle/KhPI-Press/52280>.

Література та навчальні матеріали

Основна:

1. The C++ programming language / Bjarne Stroustrup.—4th edition.- Addison-Wesley Professional, 2013. - 1368p.
 2. Object-Oriented Programming with C++/ E Balagurusamy – 8th edition.- McGraw Hill, 2020. – 656p.
 3. Об'єктно-орієнтоване програмування : підручник. У 2-х ч. Ч. 2. Об'єктно-орієнтований підхід до розробки програмного забезпечення /С. М. Алхімова. - Київ: КГП ім. Ігоря Сікорського, Вид-во «Політехніка», 2019.-1 9 2 с
 4. Основи програмування на C++: Навчальний посібник для студентів спеціальностей 113 – Прикладна математика та 122 – Комп'ютерні науки: навч. посіб./ Водка О.О., Дашкевич А.О., Іванченко К.В., Розова Л.В., Сенько А.В. – Харків: НТУ «ХПІ», 2021. – 114 с.
<http://repository.kpi.kharkov.ua/handle/KhPI-Press/52280>
 5. Основи об'єктно-орієнтованого програмування : навч. посібник /Гришанович Т. О., Глинчук Л. Я.; ВНУ імені Лесі Українки Луцьк : ВНУ імені Лесі Українки, 2022. – 120 с
- C++: The Complete Reference/ Shildt, G.- 4th Edition.-- McGraw Hill Education, 2002, - 1035p.

Додаткова:

1. Robert Lafore. Object Oriented Programming In C++, 4th edition. – Sams – 1040p.
2. Design patterns: elements of reusable object-oriented software / Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides. - Addison Wesley, 1995. – 395p.

Система оцінювання

Критерії оцінювання успішності студента та розподіл балів

Бали нараховуються за рейтингом за наступним співвідношенням:

Практична частина:

Виконання і захист індивідуальних завдань до лабораторних робіт – 60 балів (обов'язкова складова, є допуском до тесту або екзамену)

Теоретична частина:

1 вар.: Тест з теоретичної частини за матеріалами лекцій – 40 балів

2 вар.: Екзамен, який складається з теоретичних запитань – 40 балів. Допуском до екзамену є виконання індивідуальні завдання до лабораторних робіт.

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

Студент 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

28.08.2023

Завідувач кафедри
Олексій ВОДКА

28.08.2023

Гарант ОП
Оксана ТАТАРІНОВА