

Назва курсу	Об'єктно-орієнтоване програмування
Розробник (-и) силябусу	Мельников О.Ю., к. т. н., доцент
Викладач (-і)	Мельников О.Ю., к. т. н., доцент
Профайл викладача (-ів)	
Контактний телефон	+380634526352
E-mail	Melnyko@krok.edu.ua
Консультації	

1. Коротка анотація до курсу.

Метою курсу «Об'єктно-орієнтоване програмування» є формування знань студента щодо проектування та розробки програмного забезпечення із застосуванням об'єктно-орієнтованої парадигми програмування. У курсі розглядаються основні та додаткові елементи об'єктної моделі, поняття класів та об'єктів, основи об'єктно-орієнтованого аналізу та проектування мовою UML, вивчення синтаксису мов програмування C++ та Java, принципи об'єктно-орієнтованого програмування, отримання теоретичних знань та практичних навичок використання об'єктно-орієнтованої методології програмування задач, роботи у середовищі візуального програмування. Для створення об'єктно-орієнтованих комп'ютерних програм в курсі використовується мова програмування C++ та середовища розробки Dev-C++, C++ Builder.

2. Результати навчання.

Внесок дисципліни «Об'єктно-орієнтоване програмування» в формування наступних програмних результатів навчання:

Програмний результат навчання	Результат навчання з дисципліни
ПР9 Розробляти програмні моделі предметних середовищ, вибирати парадигму програмування з позицій зручності та якості застосування для реалізації методів та алгоритмів розв'язання задач в галузі комп'ютерних наук.	Студент може демонструвати вміння здійснювати об'єктно-орієнтований аналіз предметної області та подальше об'єктно-орієнтоване проектування програмного забезпечення з використанням мови UML та об'єктно-орієнтованих мов програмування

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.	Студент може розробляти програмне забезпечення із застосуванням об'єктно-орієнтованої парадигми програмування на мові програмування C++; самостійно опановувати нові мови програмування, що підтримують об'єктно-орієнтовану парадигму програмування.
--	--

Передумовою формування результатів навчання з дисципліни є володіння здобувачем на достатньому рівні наступних компетентностей:

Загальні компетентності:

ЗК1. здатність до абстрактного мислення, аналізу та синтезу;

ЗК5. здатність спілкуватися іноземною мовою;

ЗК12. здатність оцінювати та забезпечувати якість виконуваних робіт.

Спеціальні компетентності:

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;

СК21 здатність до написання дизайн-документів з використанням UML-нотації.

3. Обсяг курсу.

Вид заняття	Загальна кількість годин/ кредитів ECTS		
Загальна кількість годин / форма навчання	денна	заочна	дистанційна
лекції	28	—	—
практичні заняття	28	—	—
самостійна робота	94	—	—

4. Пререквізити.*

Алгоритмізація та програмування.

5. Технічне й програмне забезпечення/обладнання.*

Технічне обладнання: комп'ютер/ноутбук.

Програмне забезпечення:

- Операційна система: Windows 10 / 8.1 / 7
- IDE: Dev-C++, C++ Builder

6 Політики курсу.

Студенти мають дотримуватись кодексу академічної доброчесності:
[Кодекс академічної доброчесності Університету «КРОК» \(krok.edu.ua\)](http://krok.edu.ua).

7. Схема курсу

Тиждень / кіл-ть годин	Тема	Форма діяльності	Завдання	Термін виконання
1 / 2	Основні дані про мову програмування Сі	Лекція	Аналіз лекції, запитання	1 тиждень
1 / 2	Лінійний обчислювальний процес	Лабораторна робота	Звіт з виконання лабораторної роботи	
1 / 3	Обчислювальні процеси та основні типи даних	Самостійна робота	Огляд рекомендованих джерел інформації	
2 / 2	Специфіка мови програмування Сі	Лекція	Аналіз лекції, запитання	1 тиждень
2 / 2	Розгалужений циклічний обчислювальний процес	Лабораторна робота	Звіт з виконання лабораторної роботи	
3 / 3	Обчислювальні процеси та основні типи даних	Самостійна робота	Огляд рекомендованих джерел інформації	
3 / 2	Структуровані типи даних, робота препроцесора та виключні ситуації	Лекція	Аналіз лекції, запитання	1 тиждень
3 / 2	Обробка різних масивів з використанням підпрограм	Лабораторна робота	Звіт з виконання лабораторної роботи	
3 / 3	Структуровані типи даних і робота препроцесора	Самостійна робота	Огляд рекомендованих джерел інформації	
4 / 2	Об'єктна модель	Лекція	Аналіз лекції, запитання	1 тиждень
4 / 2	Обробка символічних даних	Лабораторна робота	Звіт з виконання лабораторної роботи	
4 / 3	Об'єктна модель та об'єктно-орієнтований підхід	Самостійна робота	Огляд рекомендованих джерел інформації	
5 / 2	Класи та об'єкти, об'єктно-орієнтований аналіз предметної області	Лекція	Аналіз лекції, запитання	1 тиждень
5 / 2	Робота з простими файлами	Лабораторна робота	Звіт з виконання лабораторної роботи	
5 / 3	Об'єктна модель та об'єктно-орієнтований підхід	Самостійна робота	Огляд рекомендованих джерел інформації	
6 / 2	Концептуальна модель	Лекція	Аналіз лекції, запитання	1 тиждень
6 / 2	Побудова UML-діаграм	Лабораторна робота	Звіт з виконання лабораторної роботи	
6 / 3	Діаграма варіантів використання	Самостійна робота	Огляд рекомендованих джерел інформації	
7 / 2	Логічна модель	Лекція	Аналіз лекції, запитання	1 тиждень

7 / 2	Побудова UML-діаграм	Лабораторна робота	Звіт з виконання лабораторної роботи	
7 / 3	Діаграми класів та взаємодії	Самостійна робота	Огляд рекомендованих джерел інформації	
8 / 2	Фізична модель	Лекція	Аналіз лекції, запитання	
8 / 2	Побудова UML-діаграм	Лабораторна робота	Звіт з виконання лабораторної роботи	1 тиждень
8 / 3	Діаграми реалізації	Самостійна робота	Огляд рекомендованих джерел інформації	
9 / 2	Поняття класу, конструктори, деструктори, доступність, дані та функції	Лекція	Аналіз лекції, запитання	1 тиждень
9 / 2	Перевантаження функцій	Лабораторна робота	Звіт з виконання лабораторної роботи	
9 / 3	Об'єктно-орієнтоване програмування на мові C++	Самостійна робота	Огляд рекомендованих джерел інформації	
10 / 2	Спадкування класів і додаткові питання ООП	Лекція	Аналіз лекції, запитання	1 тиждень
10 / 2	Пересування графічних об'єктів на мові C++ та побудова діаграми класів	Лабораторна робота	Звіт з виконання лабораторної роботи	
10 / 3	Об'єктно-орієнтоване програмування на мові C++	Самостійна робота	Огляд рекомендованих джерел інформації	
11 / 2	Базові конструкції мови програмування Java	Лекція	Аналіз лекції, запитання	1 тиждень
11 / 2	Створення простих програм мовою програмування Java	Лабораторна робота	Звіт з виконання лабораторної роботи	
11 / 3	Основи програмування на мові Java	Самостійна робота	Огляд рекомендованих джерел інформації	
12 / 2	Основні положення роботи в середовищі візуального програмування. Палітра компонентів	Лекція	Аналіз лекції, запитання	1 тиждень
12 / 2	Компоненти середовища візуального програмування	Лабораторна робота	Звіт з виконання лабораторної роботи	
12 / 3	Середовище візуального програмування	Самостійна робота	Огляд рекомендованих джерел інформації	
13 / 2	Особливості мови візуального програмування	Лекція	Аналіз лекції, запитання	1 тиждень
13 / 2	Розробка простого додатку у середовищі візуального програмування	Лабораторна робота	Звіт з виконання лабораторної роботи	
13 / 3	Середовище візуального програмування	Самостійна робота	Огляд рекомендованих джерел інформації	
14 / 2	Створення додатків у середовищі візуального програмування	Лекція	Аналіз лекції, запитання	1 тиждень
14 / 2	Побудова графіків функцій у середовищі візуального програмування	Лабораторна робота	Звіт з виконання лабораторної роботи	
14 / 3	Створення додатків у середовищі візуального програмування	Самостійна робота	Огляд рекомендованих джерел інформації	

8. Система оцінювання та вимоги/ Система оцінювання та критерії

Загальна система оцінювання курсу	Оцінювання проводиться за 100-бальною накопичувальною шкалою. Бали накопичуються за наступним співвідношенням: Тема 1 – 20 балів (лабораторні роботи – 10 балів, контрольні роботи – 4 бали, тестування з теорії – 6 балів); Теми 2+3 – 25 балів (лабораторні роботи – 15 балів, тестування з теорії – 5+5=10 балів); Тема 4 – 30 балів (лабораторні роботи – 20 балів, контрольна робота – 5 балів, тестування з теорії – 5 балів); Тема 5 – 25 балів (лабораторні роботи – 15 балів, контрольна робота – 5 балів, тестування з теорії – 5 балів)
Практичні заняття	Здача звітів з лабораторних робіт повинна здійснюватися відповідно до встановлених дедлайнів. Виконані завдання завантажуються студентом у відповідні модулі на платформі Moodle. Контрольні роботи та тести виконуються на платформі Moodle.
Умови допуску до підсумкового контролю	Виконання всіх контрольних робіт, тестів та лабораторних робіт є обов'язковою умовою для допуску до складання іспиту

9. Основна** та додаткова література до курсу.

Базова література

1. Мельников О. Ю. Програмування та алгоритмічні мови : посібник для студентів спеціальностей «Системний аналіз» та «Інформаційні системи та технології» / О. Ю. Мельников. – Краматорськ : ДДМА, 2018. – 236 с. – ISBN 978-966-379-870-7

2. Мельников О. Ю. Об'єктно-орієнтований аналіз і проектування інформаційних систем : посібник для студентів спеціальностей «Системний аналіз» та «Інформаційні системи та технології» / О. Ю. Мельников. – Вид. 3-є, перероб. та доп. – Краматорськ : ДДМА, 2020. – 208 с. – ISBN 978-966-379-954-4

Допоміжна література

3. Гради Буч. Объектно-ориентированный анализ и проектирование с примерами приложений на C++, 2-е изд. / Пер. с англ. – М.: «Издательство Бином», СПб.: «Невский диалект», 2001. – 560 с.

4. Кен Арнолд, Джеймс Гослинг, Дэвид Холмс. Язык программирования Java. 3-е изд.: Пер. с англ. – М.: Издательский дом «Вильямс», 2001. – 624 с.

До уваги здобувачів: усі навчально-методичні матеріали (презентації лекцій, тренувальні тести, тренувальні практичні завдання, завдання тощо) подані в Moodle за посиланням <https://dist.krok.edu.ua/user/index.php?id=20>