

Об'єктно-орієнтоване програмування	
1. Загальна інформація	
Освітня програма (галузь, спеціальність, рівень вищої освіти, форма навчання)	Освітня програма «Інженерія програмного забезпечення» Галузь знань 12 – Інформаційні технології Спеціальність 121 – Інженерія програмного забезпечення Перший (бакалаврський) рівень вищої освіти Форма навчання денна/заочна
Тип дисципліни (нормативна/вибіркова)	Нормативна
Кількість кредитів ECTS та кількість годин (лекції/ практичні / лабораторні / самостійна робота студентів), форма контролю	9 кредитів, 270 годин, Денна: 50 лекц., 50 лаб., 170 сам. роб. Заочна: 16 лекц., 16 лаб., 238 сам. роб. форма контролю – іспит, залік
Викладачі (П. І. Б, наукові ступені і звання, контактний e-mail)	Ігор КОТОВ, доцент, д-р. техн. наук (rioexito@knu.edu.ua) Антон КОЗИКОВ, старший викладач (kozikov.av@knu.edu.ua)
Посилання на матеріали дисципліни: робоча програма, методичні матеріали	http://mpz.knu.edu.ua/
Мова викладання	Українська
Інформація про розклад занять	http://asu.knu.edu.ua/time-table/chair
Кафедра: адреса, телефон, QR-code, e- mail, сайт	вул. В. Матусевича, 11, м. Кривий Ріг тел. 056-409-06-07 mpz@knu.edu.ua http://mpz.knu.edu.ua/
2. Коротка анотація до курсу	
У курсі розкриті поняття і методи використання основної методології програмування і програмної інженерії – об'єктно-орієнтованої парадигми. Об'єктно-орієнтоване програмування (ООП) сьогодні є найбільш потужним засобом, який дозволяє як моделювати об'єкти реального світу, так і генерувати віртуальні об'єкти в пам'яті комп'ютера. Використання візуального програмування істотно полегшує розробку інтерфейсу додатків, орієнтованого на стандарт візуальних оболонок. Області застосування ООП дозволяють розробляти додатки від найпростіших розрахункових форм до багатокористувацьких програмних комплексів із застосуванням графічного інтерфейсу і засобів мережі Internet. Курс розкриває такі положення ООП. Основні підходи до розробки інформаційних систем. Поняття АРМа, як професійної інформаційної системи. Програмування як методологія розробки інформаційних систем. Мови програмування - алгоритмічні, функціональні, декларативні. Розвиток парадигм програмування і їх взаємозв'язок. Основи об'єктно-орієнтованого програмування (ООП). Поняття об'єкту. Об'єктно-орієнтований аналіз та об'єктно-орієнтоване проектування.	



3. Мета та завдання курсу

Мета курсу:

це формування у студентів теоретичних знань і практичних навиків проектування, розробки та відлагодження додатків із застосуванням сучасних технологій та мов об'єктно-орієнтованого програмування, таких як C++, що забезпечують вивчення нових концепцій візуального програмування та найбільш ефективних технологій створення програмного забезпечення в процесі вирішення прикладних професійних задач; навчити студентів розробці алгоритмів на основі структурного і об'єктно-орієнтованого підходу.

Завдання курсу:

ознайомити студентів з методами структурного і об'єктно-орієнтованого програмування як найбільш поширеними і ефективними методами розробки програмних продуктів; знайомство з основними структурами даних і типовими методами обробки цих структур; навчити студентів розробці алгоритмів на основі структурного і об'єктно-орієнтованого підходу; навчити студентів критеріям перевірки правильності побудови класів, основним тенденціям в галузі розвитку технологій об'єктно-орієнтованого програмування; закріпити навички алгоритмізації та програмування на основі вивчення мов програмування Object Pascal, C++, Java; прищепити студентам знання способів використання основних візуальних компонентів розробки додатків; дати досвід розробки власних засобів контролю та управління елементами програмних систем; ознайомити студентів з принципами функціонування та управління спеціальними засобами програмування – реалізація багатозадачності, COM технології; створення практичної бази для вивчення інших навчальних дисциплін.

4. Програмні компетентності

Здатність до абстрактного мислення, аналізу та синтезу. Здатність застосовувати знання у практичних ситуаціях. Здатність до пошуку, оброблення та аналізу інформації з різних джерел. Здатність розробляти архітектури, модулі та компоненти програмних систем. Володіння знаннями про інформаційні моделі даних, здатність створювати програмне забезпечення для зберігання, видобування та опрацювання даних. Здатність реалізовувати фази та ітерації життєвого циклу програмних систем та інформаційних технологій на основі відповідних моделей і підходів розробки програмного забезпечення. Здатність до алгоритмічного та логічного мислення. Компетентності у таких областях: абстрактні тип даних; інкапсуляція, класи і об'єкти; успадкування; поліморфізм (динамічний – механізм віртуальних функцій, статичний – перевантаження функцій і операцій), здатність розуміти і самостійно розробляти алгоритми для вирішення задач системного і прикладного програмування; володіння основами об'єктно-орієнтованої мови C ++.

5. Програмні результати навчання

Знати: прийоми та засоби алгоритмізації обчислювальних задач; основні принципи об'єктно-орієнтованого програмування; призначення та особливості використання засобів інтегрованих середовищ розробки (IDE) для розробки професійних додатків; опис класів в мовах програмування OBJECT PASCAL, Java, C ++; спадкування, поліморфізм, віртуальні і динамічні методи; обробку подій і повідомлень Windows; візуальні методи розробки Windows-додатків; обробку виняткових ситуацій. Вміти: формалізувати поставлену задачу; розробити структури типів даних для представлення предметної області розв'язуваної задачі; розробляти Windows-додатки в сучасних середовищах - IDE; проектувати ієрархію класів для вирішення конкретних прикладних завдань; реалізовувати робочі проекти ієрархії класів на сучасних мовах програмування; тестувати і налагоджувати додатки з метою підвищення надійності й ефективності; організовувати управління іншими додатками, використовуючи механізми автоматизації, використовувати сучасні готові бібліотеки класів, технології та інструментальні засоби.

6. Матеріально-технічне / інформаційне та навчально-методичне забезпечення

Матеріально-технічне забезпечення (лекційні аудиторії, навчальні лабораторії, комп'ютерні класи, програмне забезпечення, мультимедійне обладнання тощо), навчально-методичне забезпечення, що дозволяє забезпечити освітній процес протягом усього терміну реалізації освітньо-професійної програми.

7. Тематика курсу				
Структура і функціонування інформаційних систем. Програмування як методологія розробки інформаційних систем. Основи об'єктно-орієнтованого програмування (ООП). Поняття об'єкту. Основи мови C ++. Базові типи даних. Змінні типу покажчик. Покажчик на об'єкт. Мова програмування C++ як мова об'єктного проектування програмного забезпечення. Структура класу. Властивості і методи класу. Конструктори класу. Динамічне розгортання об'єктів в пам'яті. Методи ініціювання та ліквідування об'єктів. Повторне використання коду. Успадкування. Конструктори і деструктори похідних класів. Перевантаження методів класу. Поліморфізм і абстракція класів. Операційна система Windows. Події в ОС Windows. Середовище візуального програмування. Стандартні компоненти. Діалогові вікна. Організація меню. Механізм реалізації подій. Графічні компоненти. Розробка систем керування базами даних. Багатопотокові програми. Основи COM-технології.				
8. Система оцінювання				
Система оцінювання за 100-бальною шкалою	Поточний контроль (за 100-бальною шкалою)			Підсумковий контроль (відповідно до 100 бальної шкали ECTS)
Виконання усіх лабораторних робіт – максимум 100 балів	Студент має вчасно виконати і захистити лабораторні роботи (ЛР) з завданнями різного рівня, набираючи бали			Залік, іспит. У разі виконання студентом усіх видів поточних контрольних заходів і складання заліку, іспиту студенту виставляється підсумкова оцінка. Підсумкова оцінка складається за 100-бальною шкалою. Студент повинен отримати не менше 50 балів.
	Вид роботи	Максимальна кількість балів		
		базовий рівень	достатній рівень	
	ЛР№ 1 – ЛР№ 10	По 6 балів	По 8 балів	
Всього	60	80	100	
9. Політика курсу				
Відвідування занять. Регуляція пропусків. Відвідування усіх занять є обов'язковим. У випадку пропуску лабораторного заняття, студент має виконати та здати лабораторну роботу згідно графіку, наведеного у робочій програмі дисципліни. У випадку пропуску лекції студент опрацьовує матеріал самостійно та може задати питання на консультації.				
Політика академічної доброчесності регламентується Положенням про академічну доброчесність у Криворізькому національному університеті .				
Використання комп'ютерів/телефонів на занятті				
На заняттях з курсу допускається використання персональних мобільних пристроїв задля досягнення навчальної мети. Є не припустимим використання персональних мобільних пристроїв задля сторонніх цілей.				

Розробники силабусу:

Доцент кафедри моделювання та програмного забезпечення,
д-р техн. наук Ігор КОТОВ

підпис

Завідувач кафедри моделювання та програмного забезпечення,
доцент, канд. пед. наук Андрій СТРЮК

підпис

Гарант ОПІ, доцент, канд. пед. наук Андрій СТРЮК

підпис