

СИЛАБУС НАВЧАЛЬНОЇ ДИСЦИПЛІНИ КРОСПЛАТФОРМНЕ ПРОГРАМУВАННЯ

1. Викладачі

Іванюк Віталій Анатолійович, доктор технічних наук, доцент, завідувач кафедри інформатики
Телефон: +38 (068) 746 33 28
e-mail: wivanyuk@kpmu.edu.ua
<https://cs.kpmu.edu.ua/2019/11/04/ivaniuk-vitalij-anatolijovych/>

Сторінка курсу в MOODLE: <https://moodle.kpmu.edu.ua/course/view.php?id=4493>

Мова викладання: українська

2. Анотація до курсу

Навчальна дисципліна присвячена розробці і використанню крос-платформних програмних систем.

3. Мета і завдання курсу

Метою навчального курсу є формування здатності аналізувати та синтезувати науково-технічну, природничо-наукову та загальнонаукову інформацію, здатності до об'єктно-орієнтованого мислення, знання об'єктно-орієнтованих мов програмування та уміння застосовувати об'єктно-орієнтований підхід під час проектуванні складних програмних систем, здатності до програмної реалізації алгоритмів розв'язання задач, розроблення системного та прикладного програмного забезпечення інформаційних систем і технологій, здатності застосовувати CASE-засоби під час проектування прикладного програмного забезпечення.

4. Результати навчання

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.

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.

Використовувати інструментальні засоби розробки клієнт-серверних застосувань, проектувати концептуальні, логічні та фізичні моделі баз даних, розробляти та оптимізувати запити до них, створювати розподілені бази даних, сховища та вітрини даних, бази знань, у тому числі на хмарних сервісах, із застосуванням мов веб-програмування.

Володіти навичками управління життєвим циклом програмного забезпечення, продуктів і сервісів інформаційних технологій відповідно до вимог і обмежень замовника, вміти розробляти проектну документацію (техніко-економічне обґрунтування, технічне завдання, бізнес-план, угоду, договір, контракт).

Застосовувати знання методології та CASE-засобів проектування складних систем, методів структурного аналізу систем, об'єктноорієнтованої методології проектування при розробці і дослідженні функціональних моделей організаційно-економічних і виробничотехнічних систем.

Виконувати паралельні та розподілені обчислення, застосовувати чисельні методи та алгоритми для паралельних структур, мови паралельного програмування при розробці та експлуатації паралельного та розподіленого програмного забезпечення.

5. Формат курсу

Очний курс з елементами дистанційного навчання в системі Moodle

6. Обсяг і ознаки курсу

Найменування показників	Характеристика навчальної дисципліни
Галузь знань	12 Інформаційні технології
спеціальність	122 Комп'ютерні науки
освітня програма	Комп'ютерні науки та інформаційні технології
Рік навчання	3
Семестр навчання	5
нормативна/вибіркова	Нормативна
Кількість кредитів ЄКТС	6
Загальний обсяг годин	180
Кількість годин навчальних занять	72
Лекційні заняття	22
Практичні заняття	0
Лабораторні заняття	50
Самостійна та індивідуальна робота	108
Форма підсумкового контролю	Екзамен

7. Пререквізити і кореквізити курсу

Для успішного вивчення навчального матеріалу достатньо компетентностей отриманих після вивчення навчальних дисциплін Програмування, Об'єктно-орієнтоване програмування, Операційні системи, Комп'ютерні мережі.

8. Технічне й програмне забезпечення, обладнання

Для проведення навчальних занять необхідна наявність мультимедійного забезпечення.
Лабораторія обчислювальної техніки.
Програмне забезпечення: IntelliJ.

9. Політика курсу

Увесь навчальний контент розміщено в модульному середовищі навчання К-ПНУ імені Івана Огієнка – moodle. Підготовка та виконання завдань лабораторних робіт є обов'язковими для кожного студента.

Академічна доброчесність. Очікується, що роботи студентів будуть їх оригінальними дослідженнями чи міркуваннями. Відсутність посилань на використані джерела, фабрикування джерел списування, втручання в роботу інших студентів становлять приклади можливої академічної недоброчесності. Виявлення ознак академічної недоброчесності в роботі студента є підставою для її незарахування викладачем, незалежно від масштабів плагіату чи обману.

Відвідання занять. Очікується, що всі студенти відвідають усі лекції та лабораторні заняття курсу. Студенти мають інформувати викладача про неможливість відвідати заняття. Студенти зобов'язані дотримуватися термінів виконання усіх видів робіт, передбачених курсом. Форми поточного та підсумкового контролю: поточний контроль реалізується на практичних заняттях; підсумковий контроль проводиться у формі екзамену.

10. Програма навчальної дисципліни

	Змістовий модуль 1. Теоретичний модуль
1	Вступне заняття
2	Концепції та принципи крос-платформного програмування
3	Компоненти та інтерфейси
4	Проектування компонентних систем. Механізми взаємодії компонентів
5	Архітектура компонентних систем. Розподілені компонентні системи
6	Проміжне програмне забезпечення. Призначення та методи використання проміжного програмного забезпечення
7	Особливості компонентних технологій. Огляд та порівняння компонентних технологій

8	Агентно-орієнтовані технології програмування. Принципи організації та застосування агентно-орієнтованих систем
9	Програмування сервісів хмарних обчислень. Організація та класифікація систем хмарних обчислень
10	Експлуатація крос-платформних програмних систем. Забезпечення функціональності та супроводу крос-платформних систем
11	IntelliJ IDEA
	Змістовий модуль 2. Практична частина
1	ВСТУП В ООП І КЛАСИ
2	ТИПИ ДАНИХ І ОПЕРАТОРИ
3	КЛАСИ І ОБ'ЄКТИ
4	УСПАДКУВАННЯ ТА ПОЛІМОРФІЗМ
5	ВНУТРІШНІ КЛАСИ
6	ІНТЕРФЕЙСИ І АНОТАЦІЇ
7	РЯДКИ
8	ВИКЛЮЧЕННЯ ТА ПОМИЛКИ
9	ПОТОКИ ВВЕДЕННЯ / ВИВЕДЕННЯ
10	КОЛЕКЦІЇ
11	ПОТОКИ ВИКОНАННЯ
12	JDBC – Java DataBase Connectivity
13	Мережеві програми
14	XML & JAVA
15	Сервлети

11. Система оцінювання та вимоги

Розподіл балів між поточним, модульним та підсумковим контролем подано у таблиці.

Модуль	100
Змістовий модуль 1	0
Змістовий модуль 2	60
Поточний контроль 2	60
Екзамен	40

Курс складається з 2-х навчальних (змістовних) модулів.

Поточний контроль полягає в перевірці теоретичних знань та практичних умінь і навичок під час лабораторних занять.

Результати перевірки самостійної роботи, в тому числі виконання домашніх завдань, студентів також входять до поточного контролю. Максимальний бал оцінки поточної успішності студентів на лабораторних заняттях приймається рівним 12.

Підсумковий контроль проводиться у формі екзамену. Кожне завдання екзамену оцінюється певною кількістю балів, а загальна оцінка за екзамену формується як сума цих балів.

Рейтингова оцінка студентам виставляється відповідно до Положення про рейтингову систему оцінювання навчальних досягнень здобувачів вищої освіти Кам'янець-Подільського національного університету імені Івана Огієнка після проведення всіх навчальних занять та контрольних заходів з навчальної дисципліни.

Студенти, які мають академічну заборгованість за результатами підсумкового контролю, зобов'язані ліквідувати її в терміни, визначені графіком ліквідації академічної заборгованості.

12. Рекомендована література

Основна література:

1. Блинов, И.Н., Романчик, В. С. Java. Методы программирования : уч.-мет. пособие / И. Н. Блинов, В. С. Романчик. — Минск : издательство «Четыре четверти», 2013. — 896 с.
2. Кулямин В. В.. Технологии программирования. Компонентный подход. М. Интернет-университет информационных технологий — БИНОМ. Лаборатория знаний, 2007. — 316 с. — Режим доступа: <http://panda.ispras.ru/~kuliamin/lectures-sdt/sdt-book-2006.pdf>.
3. Лаврищева К.М. Програмна інженерія. – К.: Академперіодика, 2008. – 319 с.
4. Степанов Е.О. Кросс-платформенные и многозвенные технологии. / Интернет-университет информационных технологий - ИНТУИТ.ру, 2010 – Режим доступа: <http://www.intuit.ru/department/se/crosspl/>
5. Таненбаум Э., ван Стеен М. Распределенные системы. Принципы и парадигмы. СПб.: Питер, 2003. – 877 с.

6. Эккель Брюс. Философия JAVA. 4-е полное издание. – Спб. : Питер, 2019. 1168 с.

Додаткова література:

7. Java Agent Development Framework (JADE): Офіційний сайт програмного середовища розробки. – Режим доступу: <http://jade.tilab.com/>
8. Бишоп Д. Эффективная работа: Java 2 : программирование. Эффективная работа. - Одесса : BHV, 2002. - 592 с.
9. Брауде Эрик Дж. Технология разработки программного обеспечения – Спб.: Питер, 2004. – 655 с.
10. Васильев А.Н. Java. Объектно-ориентированное программирование: Учебное пособие : учебн.-пособ. - Санкт-Петербург : Питер, 2011. - 400 с.
11. Гультяев А. К. Виртуальные машины. Несколько компьютеров в одном. — Спб.: Питер, 2006. — 224 с.
12. Кунегин С.В. Технология CORBA, 2001 – Режим доступу: <http://kunegin.narod.ru/ref3/corba5/index.htm>
13. Медведев В.И. Особенности объектно ориентированного программирования на C++CLI и Java. - Казань : Школа, 2010. - 444 с.
14. Михеев М. О. Администрирование VMware vSphere 4.1. – М.: ДМК Пресс, 2011. – 448 с.
15. Михеев М.О. VMware Workstation 8 Network – Режим доступу: <http://www.vm4.ru/2011/10/vmware-workstation-8-network.html>
16. Монахов Вадим Язык программирования Java и среда NetBeans, 3-е издание. — Спб.: «БХВ-Петербург», 2011. — 704 с.
17. Нейгел К., Ивьян Б. и др. C# 2005 и платформа .Net 3.0 для профессионалов. –М.: ООО “И.Д. Вильямс”, 2008. –1376 с.
18. Филлипс Б., Стюарт К., Марсикано К. Android. Программирование для профессионалов. 3-е изд. — Спб.: Питер, 2017. — 688 с.
19. Швецов А. Н. Агентно-ориентированные системы: от формальных моделей к промышленным приложениям. // Всероссийский конкурсный отбор обзорно-аналитических статей по приоритетному направлению «Информационно-телекоммуникационные системы». – 2008. – 101 с. – Режим доступу: http://window.edu.ru/window/library?p_rid=56179.
20. Шилдт, Герберт, Холмс, Джеймс. Искусство программирования на Java. - Москва : Вильямс, 2005. - 336 с.

Рекомендовані джерела інформації

1. MOODLE К-ПНУ імені ІВАНА ОГІЄНКА. – <http://moodle.kpnu.edu.ua/>