

Факультет комп'ютерно-інформаційних технологій

Затверджую:
Декан ФКП
Чолишкіна С.
“20” серпня 2017

Протокол № 12 від 20 серпня 2020 р.

Завідувач кафедри Кавун С.В.



Крос-платформене програмування

спеціалізації: _____
(за наявності) _____ (назва спеціалізації)

Київ МАУП 2020

Розробник (-и) си́лабусу навчальної дисципліни:

Рябокінь Юлія Миколаївна, кандидат технічних наук, доцент, доцент кафедри комп'ютерних інформаційних систем та технологій

Викладач:

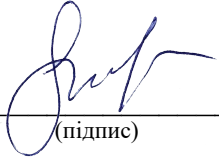
Рябокінь Юлія Миколаївна, кандидат технічних наук, доцент, доцент кафедри комп'ютерних інформаційних систем та технологій

Си́лабус розглянуто і затверджено на засіданні кафедри КІСТ
Протокол від 20 серпня 2020 № 12

Завідувач кафедри  Кавун С.В.
(підпис)

Си́лабус погоджено з гарантом (керівником) освітньої програми 122
Комп'ютерні науки
(шифр, назва освітньої програми)

20.08. 2020 р.

Керівник (гарант) освітньої програми 
(підпис)

Пролонговано:

на 20__/20__ н.р. _____ (_____), «__» ____ 20__ р., протокол № ____
(підпис) (ПІБ)

на 20__/20__ н.р. _____ (_____), «__» ____ 20__ р., протокол № ____
(підпис) (ПІБ)

на 20__/20__ н.р. _____ (_____), «__» ____ 20__ р., протокол № ____
(підпис) (ПІБ)

на 20__/20__ н.р. _____ (_____), «__» ____ 20__ р., протокол № ____

ПрАТ «ВНЗ «Міжрегіональна Академія управління персоналом» <http://maup.com.ua/>
Факультет комп'ютерно-інформаційних технологій <http://itmaup.com.ua>
Кафедра комп'ютерних інформаційних систем та технологій
http://itmaup.com.ua/?page_id=589

Назва дисципліни	<i>Крос-платформене програмування</i>
Викладач (-і)	<i>Рябокін Ю.М.</i>
Портфоліо викладача (-ів)	<i>http://itmaup.com.ua/?page_id=589</i>
Контактний тел.	<i>+380665475370</i>
E-mail:	
Сторінка дисципліни на сайті	<i>http://itmaup.com.ua/?page_id=1725</i>
Консультації	<i>Ср: 11:30-12:50 корпус 24, аудиторія 11</i>

1. Коротка анотація до дисципліни.

Курс призначений для вивчення актуальних методик та технологій створення програмного забезпечення, що може працювати більш ніж на одній платформі або операційній системі.

Курс надає майбутнім фахівцям теоретичні знання та практичні навички з крос-платформного програмування та дозволяє сформувати найважливіші практичні вміння з використанням актуальних підходів до розробки крос-платформних програмних засобів у різних галузях сучасного суспільства.

Вивчається одна з найпоширеніших у світі мов програмування – Java. На практичних і лабораторних заняттях створюються крос- платформні застосунки.

2. Мета та завдання дисципліни.

Метою викладання навчальної дисципліни є забезпечення здобуття студентами теоретичних знань і практичних навичок компонентного програмування, принципів технології розробки крос-платформних програмних систем, принципів використання засобів крос-платформного програмування.

Цілі курсу:

- ознайомлення з архітектурами та стандартами компонентних моделей;
- ознайомлення з основами взаємодії різних програмних платформ у розподілених корпоративних інформаційних системах;
- набуття знань та умінь для створення крос-платформних програмних продуктів

3. Формат курсу.

Курс передбачає теоретичні та практичні заняття.

У межах теоретичної частини забезпечуються знання:

- принципів технології розробки крос-платформних програмних систем;
- принципів використання засобів крос-платформного програмування;
- класифікації крос-платформних мов програмування;
- архітектури та стандартів компонентних моделей, комунікаційних засобів і розподілених обчислень;
- стратегій інтеграції програмних компонентів;

- основних платформ проміжного рівня та компонентні моделі;
- формальних та візуальних методів конструювання компонентів;
- поняття віртуальної машини та технології розробки програмного забезпечення для віртуальної машини Java.

Проходження практичної частини курсу формує вміння:

- розробляти вимоги та специфікації компонентів інформаційних систем і об'єктів професійної діяльності;
- проектувати компоненти програмного забезпечення;
- проектувати крос-платформний графічний інтерфейс користувача;
- реалізовувати прототипи архітектури програмного забезпечення;
- інтегрувати компоненти в систему;
- встановлювати, налаштовувати та обслуговувати системне, інструментальне і прикладне програмне забезпечення та інформаційні системи;
- використовувати базові принципи створення крос-платформних програмних систем для розроблення відповідних проєктів

4. Програмні результати навчання (інтегральні, фахові компетентності).

Результати навчання цієї дисципліни деталізують такі програмні 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.
 - Проєктувати, розробляти та аналізувати алгоритми розв'язання обчислювальних та логічних задач, оцінювати ефективність та складність алгоритмів на основі застосування формальних моделей алгоритмів та обчислюваних функцій.
 - 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.
 - Володіти навичками управління життєвим циклом програмного забезпечення, продуктів і сервісів інформаційних технологій відповідно до вимог і обмежень замовника, вміти розробляти проєктну документацію (техніко-економічне обґрунтування, технічне завдання, бізнес-план, угоду, договір, контракт).
 - Розробляти програмне забезпечення систем різних архітектур з використанням відповідних сучасних технологій, патернів та сервісів, забезпечувати розподілені обчислення та розробляти документи та форми в web, описувати та розробляти web-сервіси.
 - Забезпечувати ефективне управління якістю комп'ютерних систем на основі використання відповідних методологій, моделей, та інструментальних засобів.
-
- Застосовувати знання концепцій та методів побудови крос-платформного програмного забезпечення, володіти декількома сучасними фреймворками, включно з крос-платформним графічним інтерфейсом користувача.

4. Результати навчання.

Вивчення навчальної дисципліни передбачає формування та розвиток у студентів компетентностей:

загальних:

- здатність до абстрактного мислення, аналізу та синтезу;
- здатність застосовувати знання у практичних ситуаціях;
- знання та розуміння предметної області та розуміння професійної діяльності;

- здатність вчитися й оволодівати сучасними знаннями;
- здатність до пошуку, оброблення та аналізу інформації з різних джерел;
- здатність генерувати нові ідеї (креативність);
- здатність працювати в команді;
- здатність приймати обґрунтовані рішення;
- здатність оцінювати та забезпечувати якість виконуваних робіт.

фахових:

- здатність проєктувати 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;
- здатність застосовувати методології, технології та інструментальні засоби для управління процесами життєвого циклу інформаційних і програмних систем, продуктів і сервісів інформаційних технологій відповідно до вимог замовника;
- здатність забезпечити організацію обчислювальних процесів в інформаційних системах різного призначення з урахуванням архітектури, конфігурування, показників результативності функціонування операційних систем і системного програмного забезпечення;
- здатність забезпечувати якість комп'ютерних систем та оцінювати їх показники якості з використанням відповідних моделей та засобів на всіх етапах розробки;
- Здатність проєктувати та реалізовувати програмне забезпечення інформаційних комплексів для різноманітних операційних систем та програмних фреймворків із застосуванням сучасних бібліотек крос-платформного програмування, проводити тестування на декількох апаратних платформах, впроваджувати і підтримувати роботу інформаційних систем на сучасних платформах.

5. Опис навчальної дисципліни

Найменування показників	Характеристика дисципліни за формами навчання	
	денна	заочна
<i>Крос-платформене програмування</i>		
Курс	3	3
Семестр	5	5
Обсяг кредитів	4	4
Обсяг годин, в тому числі:	120	120
Аудиторні	50	8
Модульний контроль	2	-
Семестровий контроль	2	2
Самостійна робота	70	112
Форма семестрового контролю	<i>Іспит</i>	<i>Іспит</i>

6. Статус дисципліни: обов'язкова

7. Пререквізити – вивчення дисципліни ґрунтується на знаннях дисципліни теорія алгоритмів та математична логіка, іноземна мова.

8. Технічне й програмне забезпечення /обладнання -

Для навчального процесу потрібно: браузер, редактор перегляду pdf-файлів Foxit Reader (<https://www.foxitsoftware.com/pdf-reader/>). В якості редактора для розробки веб-платформних застосунків можна використовувати будь-який безкоштовний продукт, наприклад, Eclipse (www.eclipse.org), IntelliJ IDEA Community Edition (<https://www.jetbrains.com/idea/>), Visual Studio Code (code.visualstudio.com), NetBeans (netbeans.org), можна використовувати онлайн – середовища, наприклад, JDoodle (<https://www.jdoodle.com/online-java-compiler>), Codiva (<https://www.codiva.io>), OnlineGDB (<https://www.onlinegdb.com>), JSFiddle (<https://jsfiddle.net/>), Repl (<https://repl.it>). Ознайомитися із наявними в Інституті програмами і сервісами можна: ауд. №5: Тищенко Максим Вадимович.

9. Політика курсу - Для успішного проходження курсу та складання контрольних заходів необхідним є виконання таких обов'язків:

- не запізнюватися на заняття;
- не пропускати заняття (як лабораторні, практичні так і лекційні), у разі хвороби мати довідку або її ксерокопію;
- самостійно опрацьовувати весь лекційний матеріал та ресурси для самостійної роботи;
- конструктивно підтримувати зворотній зв'язок з викладачем на всіх етапах проходження курсу (особливо під час виконання курсового проєкту);
- своєчасно і самостійно виконувати всі передбачені програмою лабораторні та практичні завдання;
- не користуватися мобільним телефоном під час аудиторних занять;
- брати очну участь у контрольних заходах;
- будь-яке відтворення результатів чужої праці (виключаючи практичну роботу над командним проєктом), в тому числі використання завантажених з Інтернету матеріалів, як власних результатів кваліфікується як порушення норм і правил академічної доброчесності та передбачає притягнення до відповідальності у порядку, визначеному чинним законодавством

10. Схема курсу (Навчально-методична карта дисципліни)

НАВЧАЛЬНО-МЕТОДИЧНА КАРТА ДИСЦИПЛІНИ «Крос-платформене програмування»
Разом: 120 год., з них 34 год. – лекції, 17 год. – практичні заняття; самостійна робота – 70 год.

Кількість балів за семестр	100			
Модулі	Змістовий модуль I			
Назва модуля	Базові концепції кросплатформного програмування. Основи Java. Об'єктно-орієнтоване програмування мовою Java			
Лекції	1	2	3	4
Теми лекцій	Тема 1. Визначення кросплатформності. Приклади кросплатформного програмного забезпечення. Емулятори. Рівні кросплатформності: апаратний / програмний, компіляції / виконання. Основи програмування мовою Java	Тема 2. Особливості об'єктно - орієнтованого програмування на Java. Класи. Об'єкти. Наслідування. Поліморфізм. Пакети. Виняткові ситуації	Тема 3. Класи-обгортки. Інтерфейси. Колекції	Тема 4. Рядкові класи String, StringBuilder, StringBuffer у мові Java
Практичні заняття	1	2	3	4
Теми практичних занять	Створення консольних застосунків мовою Java.	Створення ієрархії класів.	Використання колекцій.	Створення проєктів з опрацюванням рядків.
Поточний контроль	модульна контрольна робота №1 (20 балів)			
Модулі	Змістовий модуль II			

Назва модуля	Створення крос-платформних застосунків із графічним інтерфейсом		
Лекції	5	6	7
Теми лекцій	Тема 5. Графічний інтерфейс користувача. Swing	Тема 5. Графічний інтерфейс користувача. Swing	Тема 7. Файлові потоки Java
Практичні заняття	5	6	7
Теми практичних занять	Створення застосунку з графічним інтерфейсом користувача	Створення графічного застосунку. Опрацювання подій миші та клавіатури	Створення застосунку для опрацювання файлових потоків
Поточний контроль	модульна контрольна робота №1 (20 балів)		
Підсумковий контроль	Іспит		

11. Система оцінювання навчальних досягнень

Вид діяльності студента	Максимальна кількість балів за одиницю	Модуль 1		Модуль 2	
		кількість одиниць	максимальна кількість балів	кількість одиниць	максимальна кількість балів
Відвідування лекцій	1	4	4	3	3
Відвідування практичних занять	1	4	4	3	3
Виконання практичних завдань	10	4	40	3	30
Виконання завдань для самостійної роботи (додавати завдання за потребою)	5	1	5	1	5
Виконання модульної роботи	20	1	20	1	20
Разом за кожним модулем	-		73	-	61
Максимальна кількість балів					134
134:100=1,34. Студент набрав X балів; Розрахунок: X:1,34 = загальна кількість балів.					
Усього за навчальною дисципліною					100

12. Завдання для самостійної роботи та критерії її оцінювання

№ з/п	Зміст самостійної роботи студента	Обсяг СРС (годин)
1.	Опрацювання лекційного матеріалу	26
2.	Підготовка до практичних занять	36
3.	Підготовка до модульних контрольних робіт №1, №2	8
Усього за навчальною дисципліною		70

Критерії оцінювання:

- змістовність – 1 бал
- відповідність темі та стилю оформлення – 1 бал

13. Форми проведення модульного контролю та критерії оцінювання

Модульний контроль здійснюється шляхом написання модульної контрольної роботи.

Студентами пропонуються варіанти МКР, кожен білет містить по 2 теоретичних запитання та 2 практичних завдання.

Критерії оцінювання:

- виконання МКР – максимум 20 балів.

Критеріями оцінки правильності виконання модульних контрольних завдань є:

№ п/п	Критерії оцінки знань студента виявлених під час проведення модульного контролю	кількість балів
-------	---	-----------------

1.	студент в процесі письмової відповіді дає правильні відповіді на всі поставлені запитання , виявляє високий рівень знань теоретичного та практичного матеріалу. Викладає свою відповідь системно та логічно, упевнено і правильно аргументує власну позицію, робить висновки, тощо.	20 балів
2.	студент має належний рівень знань теоретичного та практичного матеріалу, на поставлені запитання відповіді дає, переважно, правильні, однак допускає певні неточності у визначеннях, не завжди належно (коректно) аргументує відповідь або правильно відповідає лише на половину поставлених запитань , тощо.	15 балів
3	студент має задовільний рівень знань теоретичного та практичного матеріалу, на поставлені запитання відповідає, але не на всі, допускає певні неточності у визначеннях базових категорій, не завжди належно (коректно) аргументує або правильно дає відповідь на 1/3 (одну третину) поставлених запитань тощо.	10 балів
4.	студент дає неправильні відповіді на поставлені запитання, виявляє неналежний рівень знань теоретичного та практичного матеріалу, неспроможний послідовно і правильно аргументувати свою відповідь або взагалі не в змозі відповісти на поставлені запитання , тощо.	0 балів

14. Форми проведення семестрового контролю та критерії оцінювання: *Існум*

15. Орієнтовний перелік питань для семестрового комплексного контролю.

1. Визначення кроссплатформності
2. Основи Java..
3. Приклади кроссплатформного програмного забезпечення.
4. Емулятори.
5. Рівні кроссплатформності: апаратний / програмний, компіляції / виконання.
6. Основи програмування мовою Java
7. Особливості об'єктно - орієнтованого програмування на Java.
8. Класи.
9. Об'єкти.
10. Наслідування.
11. Поліморфізм.
12. Пакети.
13. Виняткові ситуації.
14. Класи-обгортки.
15. Інтерфейси.
16. Колекції.
17. Використання колекцій.
18. Рядкові класи String, StringBuilder, StringBuffer у мові Java.
19. Графічний інтерфейс користувача. Swing
20. Основи графіки Java.
21. Опрацювання подій
22. Файлові потоки Java.
23. Об'єктно-орієнтоване програмування мовою Java
24. Створення застосунку для опрацювання файлових потоків
25. Створення застосунку з графічним інтерфейсом користувача.
26. Опрацювання подій миші та клавіатури.

16. Шкала відповідності оцінок

Сума балів за всі види навчальної діяльності	Оцінка ECTS	Оцінка за національною шкалою	
		для екзамену, курсового проекту (роботи), практики	для заліку
90 – 100	A	відмінно	зараховано
82-89	B	добре	
75-81	C		
68-74	D	задовільно	
60-67	E		
35-59	FX	незадовільно з можливістю повторного складання	не зараховано з можливістю повторного складання
1-34	F	незадовільно з обов'язковим повторним вивченням дисципліни	не зараховано з обов'язковим повторним вивченням дисципліни

17. Рекомендовані джерела (література):

1. Костенко А. В. Крос-платформне програмування : навч. посіб. / А. В. Костенко, В. С. Костирко, М. І. Плеша, Львів. торг.-екон. ун-т; Відп. за вип. А. В. Костенко. Львів : Вид-во Львів. торг.-екон. ун-ту, 2019. 247 с.
2. Арнольд К., Гослинг Д. Язык программирования Java. СПб.: Питер, 2002. 250 с.
3. Дейтел М., Дейтел Дж., Сатори М. Технологии программирования на Java2. Книга 1. Графика. Интерфейс пользователя. М. Бином, 2003.
4. Дейтел М., Дейтел Дж., Сатори М. Технологии программирования на Java2. Книга 2. Распределенные приложения. М. Бином, 2003.
5. Дейтел М., Дейтел Дж., Сатори М. Технологии программирования на Java2. Книга 3. Корпоративные сервлеты. М. Бином, 2003.
6. Ноутон П, Шилдт Г. Java2. - С.-Пб.: БХВ-Санкт-Петербург, 2001.
7. Портякин И. Swing: эффектныe пользовательские интерфейсы. С.-Пб.: Питер, 2011. URL : https://ipsoftware.ru/books/swing_book_2/
8. Скотт К. Java для студента. СПб.: БХВ-Петербург, 2007. 448 с.; ил.
9. Флэнаген Д. Java в примерах: Справочник. 2е издание. Спб.: Символ-Плюс, 2003. 664 с.
10. Хорстман К, Корнелл Г. Java2. Основы. Библиотека профессионала. Том 1. М. Вильямс, 2007.
11. Хорстман К, Корнелл Г. Java2. Тонкости программирования. Библиотека профессионала. Том 2. М. Вильямс, 2007.
12. Эккель Б. Философия Java. 4-е полное изд. СПб.: Питер, 2015.

1168 с. URL : <https://www.twirpx.com/file/2291969/> .

13. Java® Notes for Professionals. URL :
<https://books.goalkicker.com/JavaBook/> .

Електронні джерела:

1. The Java Tutorials. – <http://docs.oracle.com/javase/tutorial/>
2. Java™ Platform, Enterprise Edition 6 API Specification. – <http://docs.oracle.com/javaee/6/api/>

