

Міністерство освіти і науки України
Національний аерокосмічний університет ім. М. Є. Жуковського
«Харківський авіаційний інститут»

Кафедра комп'ютерних систем, мереж і кібербезпеки (№ 503)

ЗАТВЕРДЖУЮ

Голова НМК



(підпис)

М.С. Зряхов

(ініціали та прізвище)

«30» серпня 2019 р.

**РОБОЧА ПРОГРАМА ВИБІРКОВОЇ
НАВЧАЛЬНОЇ ДИСЦИПЛІНИ**

Кросплатформенні технології

(назва навчальної дисципліни)

Галузь знань: 12 «Інформаційні технології»
(шифр і найменування галузі знань)

Спеціальність: 123 «Комп'ютерна інженерія»
(код і найменування спеціальності)

Освітня програма: «Комп'ютерні системи та мережі»,
«Системне програмування»
(найменування освітньої програми)

Форма навчання: денна

Рівень вищої освіти: перший (бакалаврський)

Харків 2019 рік

Робоча програма «Кросплатформенні технології»

(назва дисципліни)

для студентів за спеціальністю 123 «Комп'ютерна інженерія»
освітніми програмами «Комп'ютерні системи та мережі», «Системне програмування»

«26» серпня 2019 р., – 11 с.

Розробники: Нікітіна Тетяна Сергіївна, доцент кафедри 503

(автор, посада, науковий ступень та вчене звання)



(підпис)

Стрелкіна Анастасія Андріївна, асистент кафедри 503

(автор, посада, науковий ступень та вчене звання)



(підпис)

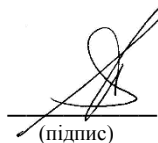
Робочу програму розглянуто на засіданні кафедри комп'ютерних систем, мереж і кібербезпеки

(назва кафедри)

Протокол № 1 від «30» серпня 2019 р.

Завідувач кафедри д.т.н., професор

(науковий ступінь і вчене звання)



(підпис)

В.С. Харченко

(ініціали та прізвище)

1. Опис навчальної дисципліни

Найменування показника	Галузь знань, спеціальність, освітня програма, рівень вищої освіти	Характеристика навчальної дисципліни (денна форма навчання)
Кількість кредитів – 4,5	<p>Галузь знань 12 «Інформаційні технології»</p> <p>Спеціальність 123 «Комп'ютерна інженерія»</p> <p>Освітні програми «Комп'ютерні системи та мережі», «Системне програмування»</p> <p>Рівень вищої освіти: перший (бакалаврський)</p>	Цикл професійної підготовки за вибором
Кількість модулів – 1		Навчальний рік
Кількість змістовних модулів – 2		2019/2020
Індивідуальне завдання: <i>немає</i>		Семестр
Загальна кількість годин – 64*/135		7-й
		Лекції*
		32 години
		Практичні, семінарські*
		<u> 0 </u> годин
		Лабораторні*
	32 години	
	Самостійна робота	
	71 година	
	Вид контролю	
	модульний контроль, іспит	
Кількість тижневих годин для денної форми навчання: аудиторних – 4, самостійної роботи студента – 4,4		

Співвідношення кількості годин аудиторних занять до самостійної роботи становить: 64/71.

* Аудиторне навантаження може бути зменшене або збільшене на одну годину залежно від розкладу занять.

2. Мета та завдання навчальної дисципліни

Мета вивчення: (ВБ6) надання студентам основних положень та парадигм кросплатформних технологій з відповідними моделями, методами та алгоритмами для створення сучасних програмних продуктів. А також оволодіти навичками й засобами розробки кросплатформного програмного забезпечення. Придбати навички й уміння по створенню кросплатформного програмного забезпечення на програмній платформі Java, та його розгортання й відлагодженню. Ознайомитись з іншими платформами та засобами для створення кросплатформного програмного забезпечення: Python, Perl та Ruby.

Завдання: (ВБ6) вивчення моделей, методів кросплатформних технологій для створення програмного забезпечення комп'ютерних систем, а також:

- вивчення основних принципів побудови мультиплатформного, кросплатформного та пересувного ПЗ, необхідність у створенні платформонезалежного ПЗ;

- вивчення методів забезпечення платформонезалежності програмного забезпечення, технологій віртуалізації – VMWare, VirtualBox, платформонезалежності, які засновані на використанні скриптових мов програмування;

- вивчення технологій забезпечення платформонезалежності ПЗ, які засновані на використанні проміжної мови – Java, .Net.;

- вивчення кросплатформних інструментальних засоби та бібліотеки;

- вивчення історії розвитку платформи Java, принципів створення платформонезалежного ПЗ на платформі Java, інструментальних засоби для розробки ПЗ на платформі Java 8, його тестування та розгортання;

- вивчення типів даних, змінних та масивів, приведення типів, операцій та їх пріоритетів, операторів, методів;

- вивчення класів та інтерфейсів, методів в класах, екземплярів класів та метода new, конструкторів;

- вивчення методів прибирання сміття, перевантаження методів та конструкторів, керування доступом;

- вивчення вкладених та внутрішніх класи, наслідування, поліморфізму, абстрактних класів, пакетів та інтерфейсів;

- вивчення основ обробки виключень, типів виключень, операторів мови Java для роботи з виключеннями, вбудовані виключення, створення власних типів виключень.

- вивчення моделі потоків на платформі Java;

- вивчення оболонків типів, аутопокування, анотації;

- вивчення потоків введення-виведення (java.io), пакета java.lang, java.util;

- вивчення особливостей розробки програмного забезпечення для роботи з мережами;

- вивчення бібліотеки для побудови графічних інтерфейсів (AWT, Swing).

Програмні компетентності. Дисципліна має допомогти сформувати у студентів такі компетентності:

- (ЗК2) здатність вчитися і оволодівати сучасними знаннями;
- (ФК2) здатність використовувати сучасні методи і мови програмування для розроблення алгоритмічного та програмного забезпечення;
- (ФК3) здатність створювати системне та прикладне програмне забезпечення комп'ютерних систем та мереж;
- (ФК13) здатність вирішувати проблеми у галузі комп'ютерних та інформаційних технологій, визначати обмеження цих технологій;
- (ФК15) здатність аргументувати вибір методів розв'язування спеціалізованих задач, критично оцінювати отримані результати, обґрунтовувати та захищати прийняті рішення.

Програмні результати навчання. В результаті вивчення дисципліни студенти мають досягти такі програмні результати навчання:

- (ПРН3) знати новітні технології в галузі комп'ютерної інженерії;
- (ПРН10) вміти розробляти програмне забезпечення для вбудованих і розподілених застосунків, мобільних і гібридних систем, розраховувати, експлуатувати, типове для спеціальності обладнання.

Крім того, в результаті вивчення дисципліни студенти повинні бути здатними до рішення задач з використанням принципів кросплатформного програмування.

Міждисциплінарні зв'язки: Дисципліна базується на знаннях, отриманих під час вивчення дисциплін у циклі професійної підготовки, передбачених навчальним планом спеціальності.

Матеріал дисципліни базується на знаннях, отриманих під час вивчення дисципліни (ВБ8) «Курс на вибір 2 (Компонентно-орієнтоване програмування)», із циклу професійної підготовки.

Матеріал, засвоєний під час вивчення цієї дисципліни, є базою для дисциплін із циклу професійної підготовки, а саме (ОК7) «Інженерія програмного забезпечення», (ОК22) «Паралельні та розподілені обчислювання», ОК29 «Дипломна робота (проект) бакалавра».

3. Програма навчальної дисципліни

Модуль 1.

Змістовний модуль 1. Основи технологій кросплатформного програмування.

Тема 1. Вступ до навчальної дисципліни «Кросплатформні технології».

Предмет, мета вивчення і задачі дисципліни. Структура та зміст дисципліни і методичні рекомендації щодо її вивчення. Місце дисципліни у навчальному процесі. Вимоги до знань та вмінь тих, хто навчається. Характеристика рекомендованих під час вивчення дисципліни джерел інформації. Принцип оцінювання по модульно-рейтинговій системі. Визначення мультиплатформного, кросплатформного та пересувного ПЗ. Необхідність у створенні платформонезалежного ПЗ.

Тема 2. Принципи та технології забезпечення кросплатформенності програмного забезпечення.

Методи забезпечення платформонезалежності програмного забезпечення. Огляд технологій віртуалізації – VMWare, VirtualBox. Огляд технологій платформонезалежності, які засновані на використанні скриптових мов програмування. Технології забезпечення кросплатформенності ПЗ, які засновані на використанні проміжної мови — Java, .Net.

Тема 3. Інструментальні засоби та бібліотеки для створення кросплатформеного ПЗ.

Інструментальні засоби для створення кросплатформеного програмного забезпечення. Класифікація методів та засобів забезпечення кросплатформенності ПЗ. Мови програмування для платформонезалежного ПЗ. Кросплатформені інструментальні засоби та бібліотеки. Бібліотека wxWidgets.

Модульний контроль.

Змістовний модуль 2. Основи технологій кросплатформеного програмування на базі платформи Java.

Тема 4. Введення у програмну платформу Java.

Історія розвитку платформи Java. Огляд мови Java. Принципи створення платформонезалежного ПЗ на платформі Java 8. Види платформонезалежного програмного забезпечення. Інструментальні засоби для розробки ПЗ на платформі Java 2, його тестування та розгортання. Типи даних, змінні та масиви. Приведення типів. Операції та їх пріоритети. Оператори. Методи.

Тема 5. Об'єктно-орієнтоване програмування на платформі Java.

Знайомство з класами та інтерфейсами. Методи в класах. Екземпляри класів та метод new. Конструктори. Прибирання сміття. Перевантаження методів та конструкторів. Рекурсія. Керування доступом. Вкладені та внутрішні класи. Наслідування. Поліморфізм. Абстрактні класи. Пакети та інтерфейси.

Тема 6. Обробка виключень, багатопотокове програмування та метадані на платформі Java.

Основи обробки виключень. Типи виключень. Оператори мови Java для роботи з виключеннями. Вбудовані виключення. Створення власних типів виключень. Модель потоків на платформі Java. Головний потік. Створення потоків. Пріоритети потоків та синхронізація. Міжпотоківі комунікації. Перечислення. Оболонки типів. Аутопакування. Анотації.

Тема 7. Базові бібліотеки платформи Java.

Потоки введення-виведення (java.io). Обробка текстових рядків. Пакет java.lang. Пакет java.util. Розробка програмного забезпечення для роботи з мережами. Обробка подій на платформі Java. Бібліотеки для побудови графічних інтерфейсів (AWT, Swing). Регулярні вирази та інші бібліотеки.

Тема 8. Знайомство з мовами кросплатформеного програмування Python, Perl та Ruby.

Платформа платформонезалежного програмування Python, синтаксис та області застосування. Платформа платформонезалежного програмування Perl,

синтаксис та області застосування. Платформа платформонезалежного програмування Ruby, синтаксис та області застосування.

Тема 9. Знайомство з патернами програмування.

Поняття патернів програмування, їх призначення, типи та класифікація. Породжуючі, поведінкові та структурні патерни програмування.

Модульний контроль.

4. Структура навчальної дисципліни

Назва змістовного модуля і тем	Кількість годин				
	Усього	У тому числі			
		л	п	лаб.	с. р.
Модуль 1					
Змістовний модуль 1. Основи технологій кросплатформенного програмування					
Тема 1. Вступ до навчальної дисципліни «Кросплатформенні технології»	4	2			2
Тема 2. Принципи та технології забезпечення кросплатформенності програмного забезпечення	10	2		4	4
Тема 3. Інструментальні засоби та бібліотеки для створення кросплатформеного ПЗ Модульний контроль	14	6		4	4
Разом за змістовним модулем 1	28	10		8	10
Змістовний модуль 2. Основи технологій кросплатформенного програмування на базі платформи Java					
Тема 4. Введення у програмну платформу Java	14	4			10
Тема 5. Об'єктно-орієнтоване програмування на платформі Java	19	3		4	12
Тема 6. Обробка виключень, багатопотокове програмування та метадані на платформі Java	20	2		8	10
Тема 7. Базові бібліотеки платформи Java	19	3			16
Тема 8. Знайомство з мовами кросплатформеного програмування Python, Perl та Ruby	9	4			5
Тема 9. Знайомство з патернами програмування Модульний контроль	26	6		12	8
Разом за змістовним модулем 2	107	22		24	61
Усього годин	135	32		32	71

5. Теми семінарських занять

№ п/п	Назва теми	Кількість годин
1	<i>Не передбачено</i>	
	Разом	

6. Теми практичних занять

№ п/п	Назва теми	Кількість годин
1	<i>Не передбачено</i>	
	Разом	

7. Теми лабораторних занять

№ з/п	Назва теми	Кількість годин
1	Робота з інструментальними засобами пакета JDK: розробка простих додатків на мові Java.	4
2	Знайомство з середою розробки ПЗ на мові Java — Netbeans IDE 6.5, Відлагодження та профайлінг ПЗ на мові Java за допомогою інструментальних засобів Netbeans IDE	4
3	Розробка простої платформонезалежної програми на мові Java з використанням бібліотеки класів.	4
4	Розробка програми “ІНТЕЛДОМ”, яка моделює поведінку компонентів системи інтелектуальної будівлі, на мові Java з використанням об'єктно-орієнтованого підходу, з додаванням блоків обробки виключних ситуації та введення власних класів виключних ситуації і багатопоточності	8
5	Розроблення кросплатформеного клієнта для роботи з базою даних за допомогою JDBC, з використанням патернів програмування	12
	Разом	32

8. Самостійна робота

№ з/п	Назва теми	Кількість годин
1	2	3
1	1. Основні принципи побудови мультиплатформеного, кросплатформеного та пересувного ПЗ, необхідність у створенні платформонезалежного ПЗ. 2. Методи забезпечення платформонезалежності програмного забезпечення, технологій віртуалізації — VMWare, VirtualBox, платформонезалежності, які засновані на використанні скриптових мов програмування.	2
2	Інтегровані середовища розробки Netbeans і Eclipse.	4
3	1. Кросплатформені інструментальні засоби та бібліотеки. 2. Принципи створення платформонезалежного ПЗ на платформі Java, його тестування та розгортання.	6

1	2	3
4	1. Типи даних, змінні та масиви, приведення типів, операції та їх пріоритети, оператори, методи. 2. Класи та інтерфейси, методи в класах, екземпляри класів та метод new, конструктори.	10
5	1. Методи прибирання сміття, перевантаження методів та конструкторів, керування доступом. 2. Вкладені та внутрішні класи, наслідування, поліморфізм, абстрактні класи, пакети та інтерфейси.	12
6	Основи обробки виключень, типи виключень, оператори мови Java для роботи з виключеннями, вбудовані виключення, створення власних типів виключень.	10
7	1. Модель потоків на платформі Java. 2. Оболонки типів, аутопокування, анотації.	10
8	1. Потоки введення-виведення (java.io), пакетом java.lang, java.util; особливості розробки програмного забезпечення для роботи з мережами. 2. Бібліотеки для побудови графічних інтерфейсів (AWT, Swing).	6
9	Платформи платформонезалежного програмування Python, Perl, Ruby, синтаксис та області застосування.	5
10	Патерни програмування. Шаблони паралельного програмування. Шаблони архітектури системи.	6
	Разом	71

9. Індивідуальні завдання

Не передбачено.

10. Методи навчання

Проведення аудиторних лекцій, лабораторних робіт, консультацій, а також самостійна робота студентів з використанням відповідних матеріалів (п.14, 15).

11. Методи контролю

Проведення поточного контролю, електронного тестування, підсумковий контроль у вигляді іспиту.

12. Критерії оцінювання та розподіл балів, які отримують студенти

12.1. Розподіл балів, які отримують студенти (кількісні критерії оцінювання)

Складові навчальної роботи	Бали за одне заняття (завдання)	Кількість занять (завдань)	Сумарна кількість балів
Змістовний модуль 1			
Лабораторні роботи	0..10	2	0..20
Модульний контроль	0..25	1	0..25
Змістовний модуль 2			
Лабораторні роботи	0...10	3	0...30
Модульний контроль	0..25	1	0..25
Усього за семестр			0...100

Семестровий контроль (іспит) проводиться у разі відмови студента від балів поточного тестування й за наявності допуску до іспиту/заліку. Під час складання семестрового іспиту студент має можливість отримати максимум 100 балів.

Білет для іспиту/заліку складається з двох теоретичних питань і одного практичного завдання. В першому і другому питанні студент повинен продемонструвати теоретичні знання. У третьому пункті – показати навички у створенні кросплатформенного програмного забезпечення. Максимальна кількість балів за перший і другий пункт – по 25 балів за кожне питання, за третій пункт – 50 балів.

12.2. Якісні критерії оцінювання

Необхідний обсяг знань для одержання позитивної оцінки:

1. Знати ключові принципи кросплатформенного програмування.
2. Знати можливості та основні положення роботи з мовою програмування Java.
3. Знати призначення класів, абстрактних класів, інтерфейсів, делегатів на мові програмування Java.
4. Знати призначення основних вбудованих класів мови програмування Java.
5. Знати основні шаблони (патерни) проектування.

Необхідний обсяг вмінь для одержання позитивної оцінки:

1. Уміти проектувати об'єктну модель предметної області.
2. Уміти створювати кросплатформенні програми з консольним та графічним інтерфейсом для розв'язання поставленої задачі
3. Уміти використовувати патерни програмування.
4. Уміти використовувати серіалізацію та десеріалізацію.

12.3 Критерії оцінювання роботи студента протягом семестру

Задовільно (60-74). Показати мінімум знань та умінь. Захистити не менше 75% від усіх завдань лабораторних занять.

Добре (75-89). Твердо знати мінімум, захистити не менше 90% завдань лабораторних занять.

Відмінно (90-100). Здати всі контрольні точки з оцінкою «відмінно». Досконально знати всі теми та уміти їх застосовувати.

Шкала оцінювання: бальна і традиційна

Сума балів	Оцінка за традиційною шкалою	
	Іспит, диференційований залік	Залік
90 – 100	Відмінно	Зараховано
75 – 89	Добре	
60 – 74	Задовільно	
0 – 59	Незадовільно	Не зараховано

13. Методичне забезпечення

Навчально-методичний комплекс дисципліни розміщений у системі управління курсами кафедри комп'ютерних систем, мереж і кібербезпеки.

1. Система управління курсами кафедри комп'ютерних систем, мереж і кібербезпеки [Ел. ресурс]. URL: <https://moodle.csn.khai.edu/>.

14. Рекомендована література

Базова

1. Брюс Эккель. Философия Java. – П.: Питер, 5-е издание, 2019. – 640 с.
2. Bruce Eckel. Thinking in Java. – Prentice Hall, 5 edition, 2016. – 1150 p.
3. Herbert Schildt. Java a Beginner's Guide 5/E (Beginner's Guide). – McGraw-Hill Osborne Media; 5 edition, 2011. – 640 p.
4. Герберт Шилд. Полный справочник по Java. Java SE 7 Edition. – М.: Издательский дом “Вильямс”, 2012. - 1040 с.
5. Kathy Sierra, Bert Bates. Head First Java [Kindle Edition]. – O'Reilly Media; 2 edition, 2012. – 688 p.
6. Бондарев В.М. Учебное пособие по программированию на Java. – Х.: СМІТ, 2013. – 296 с.
7. Rogers Cadenhead, Laura Lemay. Java 6 in 21 Days. – Indiana: Sams Publishing, 2017. – 698 p.

Допоміжна

1. Robert C. Martin. Clean Code: A Handbook of Agile Software Craftsmanship. – Prentice Hall; 1 edition, 2013. – 464 p.
2. Brian Goetz, Tim Peierls. Java Concurrency in Practice. – Addison Wesley Professional, 2016. – 384 p.
3. Herb Schildt's. Java Programming Cookbook . – New York.: The McGraw-Hill Companies, 2018. – 524 p.
4. Э. Гамма, Р. Хелм, Р. Джонсон, Дж. Влиссидес. Приемы объектно-ориентированного проектирования. Паттерны проектирования. – П.: Питер, 2017. – 366 с.

15. Інформаційні ресурси

1. <http://docs.oracle.com/javase/tutorial/>
2. <http://developer.android.com/training/basics/firstapp/index.html>