

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

Кафедра «Інженерії програмного забезпечення» (№ 603)

ЗАТВЕРДЖУЮ

Гарант освітньої програми
 І.Б. Туркін
(ініціали та прізвище)

« 31 » 08 2021 р.

СИЛАБУС ОБОВ'ЯЗКОВОЇ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Об'єктно-орієнтоване програмування мовою Java
(назва навчальної дисципліни)

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

Спеціальність: 121 Інженерія програмного забезпечення
(код та найменування спеціальності)

Освітня програма: Інженерія програмного забезпечення
(найменування освітньої програми)

Форма навчання: денна

Рівень вищої освіти: перший (бакалаврський)

Силабус введено в дію з 01.09.2021 року

Харків – 2021 р.

Розробник: Нарожний В.В., к.т.н., доц.
(прізвище та ініціали, посада, науковий ступінь та вчене звання)


(підпис)

Силабус навчальної дисципліни розглянуто на засіданні кафедри інженерії програмного забезпечення (№ 603)

Протокол № 2 від « 31 » серпня 2021 р.

Завідувач кафедри д-р техн.наук., проф.
(науковий ступінь та вчене звання)


(підпис)

І.Б. Туркін
(ініціали та прізвище)

Погоджено з представником здобувачів освіти:

Представник студентського самоврядування в/р факультету № 6


(підпис)

Д.В. Кенорич
(ініціали та прізвище)

1. Загальна інформація про викладача



Нарожний Віталій Васильович, к.т.н., доцент. З 1999 року викладає в університеті. Розробник дисциплін:

- програмування мовою Java;
- об'єктно-орієнтоване програмування на Java;
- програмування для операційної системи Android;
- розробка застосунків для мобільних пристроїв.

Напрями наукових досліджень: можливість впровадження сучасних телекомунікаційних пристроїв для розробки систем IoT.

2. Опис навчальної дисципліни

Семестр, в якому викладається дисципліна – 4 семестр.

Обсяг дисципліни:

4 кредити ЄКТС (120 годин), у тому числі аудиторних – 64 годин, самостійної роботи здобувачів – 56 годин.

Форми здобуття освіти

Денна, дистанційна, дуальна.

Дисципліна – обов'язкова.

Види навчальної діяльності – лекції, лабораторні роботи, самостійна робота здобувача.

Види контролю – поточний, модульний та підсумковий (семестровий) контроль (іспит).

Мова викладання – українська.

Необхідні обов'язкові попередні дисципліни (пререквізити) – програмування мовою Java, Основи програмування, Основи програмної інженерії.

Необхідні обов'язкові супутні дисципліни (кореквізити) – немає.

3. Мета та завдання навчальної дисципліни

Мета

Формування у студентів умінь та навичок програмування мовою Java SE у частині об'єктно-орієнтованого програмування.

Завдання

Оволодіння студентами практичних навичок зі створення об'єктно-орієнтованого програмного забезпечення мовою програмування Java.

Згідно з вимогами освітньо-професійної програми студенти повинні досягти таких компетентностей:

Загальні компетентності:

- ЗК01. Здатність до абстрактного мислення, аналізу та синтезу
- ЗК02. Здатність застосовувати знання у практичних ситуаціях
- ЗК05. Здатність вчитися і оволодівати сучасними знаннями
- ЗК06. Здатність до пошуку, оброблення та аналізу інформації з різних

джерел

- ЗК07. Здатність працювати в команді.

Фахові компетентності:

•ФК02. Здатність брати участь у проектуванні програмного забезпечення, включаючи проведення моделювання (формальний опис) його структури, поведінки та процесів функціонування.

•ФК03. Здатність розробляти архітектури, модулі та компоненти програмних систем.

•ФК05. Здатність дотримуватися специфікацій, стандартів, правил і рекомендацій в професійній галузі при реалізації процесів життєвого циклу.

•ФК10. Здатність накопичувати, обробляти та систематизувати професійні знання щодо створення і супроводження програмного забезпечення та визнання важливості навчання протягом всього життя.

•ФК13. Здатність обґрунтовано обирати та освоювати інструментарій з розробки та супроводження програмного забезпечення.

- ФК14. Здатність до алгоритмічного та логічного мислення.

Програмні результати навчання:

•ПРН01. Аналізувати, цілеспрямовано шукати і вибирати необхідні для вирішення професійних завдань інформаційно-довідникові ресурси і знання з урахуванням сучасних досягнень науки і техніки.

•ПРН05. Знати і застосовувати відповідні математичні поняття, методи доменного, системного і об'єктно-орієнтованого аналізу та математичного моделювання для розробки програмного забезпечення.

•ПРН07. Знати і застосовувати на практиці фундаментальні концепції, парадигми і основні принципи функціонування мовних, інструментальних і обчислювальних засобів інженерії програмного забезпечення.

- ПРН09. Знати та вміти використовувати методи та засоби збору, формулювання та аналізу вимог до програмного забезпечення.
- ПРН10. Проводити передпроектне обстеження предметної області, системний аналіз об'єкта проектування.
- ПРН13. Знати і застосовувати методи розробки алгоритмів, конструювання програмного забезпечення та структур даних і знань.
- ПРН14. Застосовувати на практиці інструментальні програмні засоби доменного аналізу, проектування, тестування, візуалізації, вимірювань та документування програмного забезпечення.
- ПРН17. Вміти застосовувати методи компонентної розробки програмного забезпечення.
- ПРН18. Знати та вміти застосовувати інформаційні технології обробки, зберігання та передачі даних.

4. Зміст навчальної дисципліни

Модуль 1.

Змістовний модуль 1.

Тема 1. Об'єктно-орієнтованого програмування Java

- *Форма занять: лекція, лабораторна робота, самостійна робота.*
- *Обсяг аудиторного навантаження: 12 годин.*
- *Лабораторна робота: «Об'єктно-орієнтованого програмування. Наслідування», «Об'єктно-орієнтованого програмування. Наслідування», «Об'єктно-орієнтованого програмування. Метод».*
- *Обов'язкові предмети та засоби (обладнання, устаткування, матеріали, інструменти): персональний комп'ютер або ноутбук.*

Постановка теми. Об'єктно-орієнтованого програмування Java. Парадигми програмування. Принципи об'єктно-орієнтованого програмування (ООП). Переваги та недоліки ООП. Синтаксис опису класу і підкласу. Оголошення об'єкта. Операція new. Синтаксис опису методу. Передача параметрів, перевантаження і перевизначене методів.

- *Обсяг самостійної роботи здобувачів: 10 години.*

Опрацювання матеріалу лекцій. Формування питань до викладача. Оформлення лабораторної роботи та підготовка до її здачі

Тема 2. Особливості використання об'єктно-орієнтованого програмування.

- *Форма занять: лекція, лабораторна робота, самостійна робота.*
- *Обсяг аудиторного навантаження: 20 години.*

- *Лабораторні роботи:* «Об'єктно-орієнтованого програмування. Конструктор», «Об'єктно-орієнтованого програмування. Приведення», «Об'єктно-орієнтованого програмування. Абстрактний клас та Інтерфейс», «Об'єктно-орієнтованого програмування. Внутрішні, вкладений, анонімні класи» .

- *Обов'язкові предмети та засоби (обладнання, устаткування, матеріали, інструменти):* персональний комп'ютер або ноутбук.

Особливості використання Об'єктно-орієнтованого програмування. Конструктори класу. Параметризовані конструктори. Перевантаження конструкторів. Вкладені та внутрішні класи. Абстрактні методи і класи. Клас Object. Використання ключового слова super. Статичні методи. Метод main. Визначення пакета та інтерфейсу. Імпорт пакетів. Реалізація та застосування інтерфейсів. Змінні в інтерфейсах. Розширення інтерфейсів. Основні принципи обробки винятків. Типи виключень. Вбудовані виключення Java. Створення власних підкласів виключень.

- *Обсяг самостійної роботи здобувачів: 10 години.*

Опрацювання матеріалу лекцій. Формування питань до викладача. Оформлення лабораторних робіт та підготовка до їх здачі.

Модульний контроль 1

- *Форма занять: написання модульної роботи в аудиторії (за рішенням лектора допускається проведення у дистанційній формі).*

- *Обсяг аудиторного навантаження: за необхідністю*

- *Обов'язкові предмети та засоби (обладнання, устаткування, матеріали, інструменти):* персональний комп'ютер або ноутбук..

- *Обсяг самостійної роботи здобувачів – 5 години.*

Підготовка до модульного контролю.

Змістовний модуль 2.

Тема 3. Технології розробки застосунків мовою Java ООП.

- *Форма занять: лекція, лабораторна робота, самостійна робота.*

- *Обсяг аудиторного навантаження: 20 годин.*

- *Лабораторні роботи:* «Об'єктно-орієнтованого програмування. Виключення», «Об'єктно-орієнтованого програмування. Колекції» «Об'єктно-орієнтованого програмування. Лямбда-вирази».

- *Обов'язкові предмети та засоби (обладнання, устаткування, матеріали, інструменти):* персональний комп'ютер або ноутбук.

Динамічні масиви та графічний інтерфейс Java. Клас Math. Класи для роботи з строкою String і StringBuffer. Використання виключень. Огляд пакета Collection. Інтерфейси колекцій. Класи Collection. Колекції призначених для користувача класів. Програмування графіки та обробка подій. Введення в AWT: робота з вікнами, графікою і текстом. Компоненти бібліотеки Swing. Дерева. Таблиці.

- *Обсяг самостійної роботи здобувачів: 15 годин.*

Опрацювання матеріалу лекцій. Формування питань до викладача. Оформлення лабораторних робіт та підготовка до їх здачі.

Тема 4. Мережеві технології.

- *Форма занять: лекція, лабораторна робота, самостійна робота.*

- *Обсяг аудиторного навантаження: 12 годин.*

- *Лабораторні роботи: «Розробка застосунків мовою програмування Java. Поточність», «Розробка застосунків мовою програмування Java. Мережеві застосунки».*

- *Обов'язкові предмети та засоби (обладнання, устаткування, матеріали, інструменти): персональний комп'ютер або ноутбук.*

Поточна модель Java. Головний потік. Створення потоку. Створення множинних потоків. Пріоритети потоків. Синхронізація. Міжпоточні зв'язки. Призупинення, відновлення і зупинка потоків. Використання поточності. Система введення-виведення. Байтові потоки: Класи InputStream та OutputStream. Класи Object і Class. Класи-обгортки для примітивних типів. Основи роботи в мережі. Java і мережа. Мережеві класи і інтерфейси. Клас InetAddress. Сокети TCP / IP клієнтів. Використання URL. Клас URLConnection. Сокети TCP / IP серверів.

- *Обсяг самостійної роботи здобувачів: 6 годин.*

Опрацювання матеріалу лекцій. Формування питань до викладача. Оформлення лабораторних робіт та підготовка до їх здачі.

Модульний контроль 2

- *Форма занять: написання модульної роботи в аудиторії (за рішенням лектора допускається проведення у дистанційній формі).*

- *Обсяг аудиторного навантаження: за необхідністю*

- *Обов'язкові предмети та засоби (обладнання, устаткування, матеріали, інструменти): персональний комп'ютер або ноутбук.*

- *Обсяг самостійної роботи здобувачів – 5 години.*

Підготовка до модульного контролю.

5. Індивідуальні завдання

Не передбачено навчальним планом

6. Методи навчання

Словесні, наочні, практичні.

7. Методи контролю

Поточний контроль (теоретичне опитування й розв'язання практичних завдань), модульний контроль (тестування за розділами курсу) та підсумковий (семестровий) контроль (іспит).

8. Критерії оцінювання та розподіл балів, які отримують здобувачі

Складові навчальної роботи	Бали за одне заняття (завдання)	Кількість занять (завдань)	Сумарна кількість балів
Змістовний модуль 1			
Виконання і захист лабораторних (практичних) робіт	2...5	6	12...30
Модульний контроль	10...20	1	10...20
Змістовний модуль 2			
Виконання і захист лабораторних (практичних) робіт	2...5	6	12...30
Модульний контроль	10...20	1	10...20
Виконання і захист РГР (РР, РК)			
Усього за семестр			60...100

Прийнята шкала оцінювання

Сума балів за всі види навчальної діяльності	Оцінка для екзамену, курсового проекту (роботи), практики
90-100	відмінно
75-89	добре
60-74	задовільно
01-59	незадовільно з можливістю повторного складання

Іспит проводиться у вигляді тестування. Тест складається з 16 питань закритого типу (за правильну відповідь на одне питання здобувач отримує

5 балів) та двох питань відкритого типу (максимальна кількість балів за відповідь на одне питання – 10).

Під час складання семестрового іспиту здобувач має можливість отримати максимум 100 балів.

Критерії оцінювання роботи здобувача протягом семестру

Задовільно (60-74). Показати мінімум знань та умінь. Парадигми програмування. Принципи об'єктно-орієнтованого програмування (ООП). Переваги та недоліки ООП. Синтаксис опису класу і підкласу. Оголошення об'єкта. Операція new. Синтаксис опису методу. Передача параметрів, перевантаження і перевизначення методів. Конструктори класу. Параметризовані конструктори. Перевантаження конструкторів. Вкладені та внутрішні класи. Абстрактні методи і класи. Клас Object. Використання ключового слова super. Статичні методи. Метод main. Визначення пакета та інтерфейсу. Імпорт пакетів. Реалізація та застосування інтерфейсів. Змінні в інтерфейсах. Розширення інтерфейсів. Основні принципи обробки винятків. Типи виключень. Вбудовані виключення Java. Створення власних підкласів виключень. Використання винятків. Поточна модель Java. Головний потік. Створення потоку. Створення множинних потоків. Пріоритети потоків. Синхронізація. Міжпоточні зв'язки. Призупинення, відновлення і зупинка потоків. Використання поточності. Система введення-виведення. Байтові потоки: Класи InputStream та OutputStream. Класи Object і Class. Класи-обгортки для примітивних типів. Клас Math. Класи для роботи з строкою String і StringBuffer. Огляд пакета. Інтерфейси колекцій. Класи Collection. Колекції призначених для користувача класів. Програмування графіки та обробка подій. Введення в AWT: робота з вікнами, графікою і текстом. Використання елементів управління, менеджерів компоновання і меню AWT. Робота з зображеннями. Компоненти бібліотеки Swing. Деревя. Таблиці. Інші можливості і майбутнє Swing-технології. Кнопки. Основи роботи в мережі. Java і мережа. Мережеві класи і інтерфейси. Клас InetAddress. Сокети TCP / IP клієнтів. Використання URL. Клас URLConnection. Сокети TCP / IP серверів

Добре (75-89). Твердо знати мінімум, здати тестування та поза аудиторну самостійну роботу. Парадигми програмування. Принципи об'єктно-орієнтованого програмування (ООП). Переваги та недоліки ООП. Синтаксис опису класу і підкласу. Оголошення об'єкта. Операція new. Синтаксис опису методу. Передача параметрів, перевантаження і перевизначення методів. Конструктори класу. Параметризовані конструктори. Перевантаження конструкторів. Вкладені та внутрішні класи. Абстрактні методи і класи. Клас Object. Використання ключового слова super. Статичні методи. Метод main. Визначення пакета та інтерфейсу. Імпорт пакетів. Реалізація та застосування інтерфейсів. Змінні в інтерфейсах. Розширення інтерфейсів. Основні принципи обробки винятків. Типи виключень. Вбудовані виключення Java. Створення власних підкласів виключень. Використання винятків. Поточна модель Java. Головний потік. Створення потоку. Створення множинних потоків. Пріоритети

потоків. Синхронізація. Міжпоточні зв'язки. Призупинення, відновлення і зупинка потоків. Використання поточності. Система введення-виведення. Байтові потоки: Класи `InputStream` та `OutputStream`. Класи `Object` і `Class`. Класи-обгортки для примітивних типів. Клас `Math`. Класи для роботи з строкою `String` і `StringBuffer`. Огляд пакета. Інтерфейси колекцій. Класи `Collection`. Колекції призначених для користувача класів. Програмування графіки та обробка подій. Введення в AWT: робота з вікнами, графікою і текстом. Використання елементів управління, менеджерів компоновання і меню AWT. Робота з зображеннями. Компоненти бібліотеки Swing. Деревя. Таблиці. Інші можливості і майбутнє Swing-технології. Кнопки. Основи роботи в мережі. Java і мережа. Мережеві класи і інтерфейси. Клас `InetAddress`. Сокети TCP / IP клієнтів. Використання URL. Клас `URLConnection`. Сокети TCP / IP серверів

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

9. Політика навчального курсу

Відпрацювання пропущених занять відбувається відповідно до розкладу консультацій, за попереднім погодженням з викладачем. Питання, що стосуються академічної доброчесності, розглядає викладач або за процедурою, визначеною у Положенні про академічну доброчесність.

10. Методичне забезпечення та інформаційні ресурси

Підручники, навчальні посібники, навчально-методичні посібники, конспекти лекцій, методичні рекомендації з проведення лабораторних робіт тощо, які видані в Університеті знаходяться за посиланням:

1. Дистанційний курс дисципліни розроблено у системі дистанційного навчання Mentor, яку впроваджено в Національному аерокосмічному університеті ім. М.Є. Жуковського «ХАІ», доступ до курсу за посиланням: <https://mentor.khai.edu/course/view.php?id=242>

2. Java SDK, Standard Edition Documentation Oracle, <http://www.oracle.com/technetwork/java/javase/overview/index.html>

3. The Java Tutorials. A practical guide for programmers. Oracle. <http://docs.oracle.com/javase/tutorial/>

4. Java Resources for Developers. <http://www.oracle.com/technetwork/java/index.html>

11. Рекомендована література

Базова

1. О.М. Васильєв. Програмування мовою Java. — К.: «Навчальна книга - Богдан». ISBN: 9789661058797. 2020. - 696 с.
2. Роберт С. Мартін. Чистий код. — /Пер. з англ.— К. : "Фабула". ISBN: 9786170952851. 2019р. 416 с.
3. Патерни проєктування. Ерік Фрімен, Елізабет Робсон. /Пер. з англ.— К. : "Фабула". ISBN: 9786170961594. 2020р. 688 с.

Допоміжна

1. Эккель Б. Философия Java. Библиотека программиста. /Пер. с англ. - СПб.: «Питер», 2001. – 880 с 7
2. Ken Arnold, James Gosling, David Holmes, The Java Programming Language, Fourth Edition, Addison-Wesley Professional, 2005, ISBN 0-321-34980-6
3. "On Java 8", Bruce Eckel, MindView LLC, 2017. ISBN 978-0-9818725-2-0.
4. Effective Java: Programming Language Guide, ISBN 0-201-31005-8, 2001; second edition: ISBN 978-0-321-35668-0, 2008; third edition: ISBN 978-0134685991, 2017