



Основи розробки програмного забезпечення на платформі Java Робоча програма навчальної дисципліни (Силабус)

Реквізити навчальної дисципліни

Рівень вищої освіти	Перший (бакалаврський)
Галузь знань	12 Інформаційні технології
Спеціальність	123 Комп'ютерна інженерія
Освітня програма	Комп'ютерні системи та мережі
Статус дисципліни	Вибіркова
Форма навчання	очна(денна)/ заочна
Рік підготовки, семестр	3 курс, осінній семестр
Обсяг дисципліни	4 кредити, 120 годин
Семестровий контроль/ контрольні заходи	залік
Розклад занять	//rozklad.kpi.ua
Мова викладання	Українська
Інформація про керівника курсу / викладачів	Лектор: старший викладач, Алещенко Олексій Вадимович alexey.aleshchenko@gmail.com Лабораторні: старший викладач, Алещенко Олексій Вадимович alexey.aleshchenko@gmail.com
Розміщення курсу	//comsys.kpi.ua

Програма навчальної дисципліни

1. Опис навчальної дисципліни, її мета, предмет вивчення та результати навчання

Дисципліна "Основи розробки програмного забезпечення на платформі Java" спрямована на опанування студентами методів та засобів програмування задач довільної складності з використанням методології, що базується на об'єктно-орієнтованій парадигмі, використовуючи платформу Java. Знання основ розробки програмного забезпечення на платформі Java необхідне для створення програмного забезпечення для комп'ютерних систем, систем реального часу, інтернет-додатків, мобільних пристроїв.

Метою викладання дисципліни "Основи розробки програмного забезпечення на платформі Java" є:

ЗДАТНІСТЬ: до абстрактного мислення, аналізу та синтезу; працювати в команді; використовувати сучасні методи і мови програмування для розроблення алгоритмічного та програмного забезпечення; створювати системне та прикладне програмне забезпечення комп'ютерних систем та мереж; забезпечувати захист інформації, що обробляється в комп'ютерних та кіберфізичних системах та мережах з метою реалізації встановленої політики інформаційної безпеки; системно адмініструвати, використовувати, адаптувати та експлуатувати наявні інформаційні технології та системи; ідентифікувати, класифікувати та описувати роботу програмно-технічних засобів, комп'ютерних та кіберфізичних систем, мереж

та їхніх компонентів шляхом використання аналітичних методів і методів моделювання; проектувати, впроваджувати, адмініструвати та обслуговувати глобальні, локальні інтелектуальні програмно - конфігуровані комп'ютерні мережі; розробляти, адаптувати, використати програмне забезпечення для покращення ефективності застосування високопродуктивних комп'ютерних систем; організації обчислювальних процесів в високопродуктивних комп'ютерних системах з різною структурною організацією на основі використання новітніх технологій планування і диспетчеризації та сучасних операційних систем;

ЗНАННЯ: основ управління проектами; для ідентифікації, формулювання і розв'язування технічних задач спеціальності, використовуючи методи, що є найбільш придатними для досягнення поставлених цілей; технічних характеристик, конструктивних особливостей, призначення і правил експлуатації програмно-технічних засобів комп'ютерних систем та мереж для вирішення технічних задач спеціальності;

УМІННЯ: проведення експериментів, збирання даних та моделювання в комп'ютерних системах; застосовувати знання для ідентифікації, формулювання і розв'язування технічних задач спеціальності, використовуючи методи, що є найбільш придатними для досягнення поставлених цілей; розв'язувати задачі аналізу та синтезу засобів, характерних для спеціальності; застосовувати знання технічних характеристик, конструктивних особливостей, призначення і правил експлуатації програмно-технічних засобів комп'ютерних систем та мереж для вирішення технічних задач спеціальності; розробляти програмне забезпечення для вбудованих і розподілених застосувань, мобільних і гібридних систем, розраховувати, експлуатувати, типове для спеціальності обладнання; виконувати експериментальні дослідження за професійною тематикою; виконувати розрахунки параметрів окремих комп'ютерних систем, комп'ютерних мереж; вміти створювати та обслуговувати бази даних.

2. Пререквізити та постреквізити дисципліни (місце в структурно-логічній схемі навчання за відповідною освітньою програмою)

Необхідні дисципліни: "Програмування. Частина 1. Програмування", "Структури даних та алгоритми", "Вища математика. Частина 1", "Аналітична геометрія та лінійна алгебра".

3. Зміст навчальної дисципліни

Тема 1. Загальна характеристика мови Java. Порівняння з іншими мовами програмування.

Прості типи та основи синтаксису Java.

Тема 2. Класи та методи в Java. Пакети та інтерфейси.

Тема 3. Робота з масивами в Java.

Тема 4. Робота зі строками в Java. Класу String, String Buffer, String Builder.

Тема 5. Перевантаження методів. Перевантаження конструкторів. Управління доступом.

Статичні елементи.

Тема 6. Властивості (поля). Наслідування властивостей. Внутрішні та анонімні класи.

Тема 7. Створення багаторівневих ієрархій. Клас Object.

Тема 8. Методи. Наслідування методів. Поліморфізм. Специфікатори super та final.

Тема 9. Реалізація поліморфізму. Перевизначення методів. Раннє та пізнє зв'язування.

Тема 10. Колекції в Java. Зв'язані списки.

Тема 11. Обробка виключень та модульне тестування.

Тема 12. Управління пам'яттю, деструктори об'єктів та Garbage Collector.

Тема 13. Пакет java.util.stream.

Тема 14. Події та орієнтоване на події управління поведінкою програми.

4. Навчальні матеріали та ресурси

Базова:

1. *Java-програмування: комп'ютерний практикум [Електронний ресурс] : навч. посіб. для студ. спеціальності 122 «Комп'ютерні науки», освітньо-професійної програми «Комп'ютерний моніторинг та геометричне моделювання процесів і систем» / КПІ ім. Ігоря Сікорського ; уклад.: Ю. А. Тарнавський. – Електронні текстові дані (1 файл: 686 Кбайт). – Київ : КПІ ім. Ігоря Сікорського, 2021. – 95 с. <https://ela.kpi.ua/bitstream/123456789/41885/1/Java-programming.pdf>*
2. *Конспект лекцій з програмного модулю "Об'єктно-орієнтоване програмування" / Розробник: асистент Алещенко Олексій Вадимович. Кафедра обчислювальної техніки ФІОТ. Затверджено на засіданні кафедри (Протокол № 11 від 24 травня 2017 р) <https://comsys.kpi.ua/katalog/files/konspekt-lekciy-oor-1.pdf>*
3. *Основи розробки програмного забезпечення на платформі Java. Методичні вказівки до виконання лабораторних робіт для здобувачів ступеня бакалавра за спеціальностями: 121 Інженерія програмного забезпечення, 123 Комп'ютерна інженерія, 126 Інформаційні системи та технології всіх форм навчання / Уклад.: Алещенко О.В. – К.: КПІ ім. Ігоря Сікорського, 2022. [Електронний ресурс]. Ухвалено кафедрою обчислювальної техніки (протокол № 10 від 25.05.2022 р.). Погоджено Методичною комісією ФІОТ (протокол № 10 від 09.06.2022 р.). <https://comsys.kpi.ua/metodichni-vkazannya-po-disciplinam>.*

Додаткова:

4. *Цибульник С. О. Технології розробки програмного забезпечення-1. Комп'ютерний практикум [Електронний ресурс] : навч. посіб. для студ. спеціальності 151 «Автоматизація та комп'ютерно-інтегровані технології», освітньо-професійної програми «Комп'ютерно-інтегровані технології та системи навігації і керування» / С. О. Цибульник ; КПІ ім. Ігоря Сікорського. Київ: КПІ ім. Ігоря Сікорського, 2021. 126 с. https://ela.kpi.ua/bitstream/123456789/41516/3/Tekhnolohii-rozrobky-prohramnoho-zabezpechennia_KompPrakt.pdf*

Навчальний контент**5. Методика опанування навчальної дисципліни (освітнього компонента)****Очна форма навчання**

Назви розділів і тем	Кількість годин				
	Всього	у тому числі			
		Лекції	Практичні (семінарські)	Лабораторні роботи	СРС
Тема 1. Загальна характеристика мови Java. Порівняння з іншими мовами програмування. Прості типи та основи синтаксису Java.	9	2	-	3	4
Тема 2. Класи та методи в Java. Пакети та інтерфейси.	6	2	-	-	4
Тема 3. Робота з масивами в Java.	9	2	-	3	4
Тема 4. Робота зі строками	9	2	-	3	4

<i>в Java. Класи String, String Buffer, String Builder.</i>					
<i>Тема 5. Перевантаження методів. Перевантаження конструкторів. Управління доступом. Статичні елементи.</i>	6	2	-	-	4
<i>Тема 6. Властивості (поля). Наслідування властивостей. Внутрішні та анонімні класи.</i>	11	4	-	3	4
<i>Тема 7. Створення багаторівневих ієрархій. Клас Object.</i>	9	2	-	3	4
<i>Тема 8. Методи. Наслідування методів. Поліморфізм. Специфікатори super та final.</i>	9	2	-	3	4
<i>Тема 9. Реалізація поліморфізму. Перевизначення методів. Раннє та пізнє зв'язування.</i>	6	2	-	-	4
<i>Тема 10. Колекції в Java. Зв'язані списки.</i>	10	6	-	-	4
<i>Тема 11. Обробка виключень та модульне тестування.</i>	8	4	-	-	4
<i>Тема 12. Управління пам'яттю, деструктори об'єктів та Garbage Collector.</i>	6	2	-	-	4
<i>Тема 13. Пакет java.util.stream.</i>	7	2	-	-	5
<i>Тема 14. Події та орієнтоване на події управління поведінкою програми.</i>	7	2	-	-	5
МКР	4	-	-	-	4
Залік	4	-	-	-	4
Всього в семестрі:	120	36	-	18	66

Назви розділів і тем	Кількість годин				
	Всього	у тому числі			
		Лекції	Практичні (семінарські)	Лабораторні роботи	СРС
Тема 1. Загальна характеристика мови Java. Порівняння з іншими мовами програмування. Прості типи та основи синтаксису Java.	9	1	-	2	6
Тема 2. Класи та методи в Java. Пакети та інтерфейси.	7	1	-	-	6
Тема 3. Робота з масивами в Java.	7,25	0,25	-	1	6
Тема 4. Робота зі строками в Java. Класи String, String Buffer, String Builder.	7,25	0,25	-	1	6
Тема 5. Перевантаження методів. Перевантаження конструкторів. Управління доступом. Статичні елементи.	6,25	0,25	-	-	6
Тема 6. Властивості (поля). Наслідування властивостей. Внутрішні та анонімні класи.	7,25	0,25	-	1	6
Тема 7. Створення багаторівневих ієрархій. Клас Object.	9	1	-	2	6
Тема 8. Методи. Наслідування методів. Поліморфізм. Специфікатори super та final.	7,25	0,25	-	1	6
Тема 9. Реалізація поліморфізму. Перевизначення методів. Раннє та пізнє зв'язування.	7,25	0,25	-	-	7
Тема 10. Колекції в Java. Зв'язані списки.	8	1	-	-	7
Тема 11. Обробка виключень та модульне тестування.	7,25	0,25	-	-	7
Тема 12. Управління пам'яттю, деструктори об'єктів та Garbage Collector.	7,25	0,25	-	-	7
Тема 13. Пакет	8	1	-	-	7

<i>java.util.stream.</i>					
Тема 14. Події та орієнтоване на події управління поведінкою програми.	8	1	-	-	7
МКР	7	-	-	-	7
Залік	7	-	-	-	7
Всього в семестрі:	120	8	-	8	104

Лекційні заняття

Очна форма навчання

№ лекції	Назва теми лекції, перелік основних питань та завдання на СРС
1	<p>«Вступ в Java». Загальна характеристика мови Java. Порівняння з іншими мовами програмування. Прості типи та основи синтаксису Java.</p> <p>СРС: Частково або повністю виконати першу лабораторну роботу даного кредитного модуля на мові програмування Java.</p>
2	<p>«Вступ в Java. Продовження». Класи та методи в Java. Пакети та інтерфейси.</p> <p>СРС: Створити декілька методів в декількох різних класах, які знаходяться в різних пакетах та реалізують різні інтерфейси.</p>
3	<p>«Робота з масивами в Java». Одновимірні, багатовимірні масиви та об'єкти, що представляють масиви в Java.</p> <p>СРС: Частково або повністю виконати другу лабораторну роботу даного кредитного модуля на мові програмування Java.</p>
4	<p>«Робота зі строками в Java». Класи String, String Buffer, String Builder.</p> <p>СРС: Частково або повністю виконати третю лабораторну роботу даного кредитного модуля на мові програмування Java.</p>
5	<p>«Методи класу. Перевантаження, статичність та доступ». Перевантаження методів. Перевантаження конструкторів. Управління доступом. Статичні елементи.</p> <p>СРС: Створити декілька перевантажених методів з різними модифікаторами доступу.</p>
6	<p>«Властивості (поля) класу. Анонімні класи на прикладі сортування масиву об'єктів». Анонімні класи. Інтерфейс Comparator. Метод Compare. Сортування масиву об'єктів. Клас Arrays. Метод Sort.</p> <p>СРС: Частково або повністю виконати четверту лабораторну роботу даного кредитного модуля на мові програмування Java.</p>
7	<p>«Наслідування та управління видимістю». Властивості (поля). Наслідування властивостей. Управління областю видимості.</p> <p>СРС: Створити клас з декількома полями з різними областями видимості. Створити клас, що наслідує попередній. Спробувати доступитися в ньому до полів попереднього класу та створити власні поля з такими ж іменами.</p>
8	<p>«Ієрархії об'єктів». Створення багаторівневих ієрархій об'єктів через наслідування та/або агрегацію. Клас Object.</p> <p>СРС: Частково або повністю виконати п'яту лабораторну роботу даного кредитного модуля на мові програмування Java.</p>

9	<p>«Наслідування методів. Поліморфізм». Наслідування методів. Поліморфізм. Специфікатори <i>super</i> та <i>final</i>.</p> <p>СРС: Частково або повністю виконати шосту лабораторну роботу даного кредитного модуля на мові програмування Java.</p>
10	<p>«Реалізація поліморфізму». Реалізація поліморфізму. Перевизначення методів. Раннє та пізнє зв'язування.</p> <p>СРС: Створити клас з декількома методами. Створити клас, що наслідує попередній та перевизначає його методи. Створити екземпляр другого класу, посилання на який буде зберігати змінна першого класу. Спробувати доступитися через вказану змінну першого класу до перевизначеного методу з другого класу.</p>
11	<p>«Колекції в Java». Колекції в Java. Зв'язані списки. Інтерфейси <i>Set</i> та <i>List</i>.</p> <p>СРС: Спробувати створити власний клас, який реалізує будь-який інтерфейс колекції (без реалізації ітератора та еквівалентності).</p>
12	<p>«Колекції в Java. Продовження». Методи <i>toArray</i>. Ітератори.</p> <p>СРС: Реалізувати ітератор в рамках попереднього СРС.</p>
13	<p>«Колекції в Java. Продовження». Унікальність елементів у колекціях типу <i>Set</i>. Еквівалентність об'єктів.</p> <p>СРС: Реалізувати перевірку еквівалентності окремих елементів та всієї колекції в рамках попереднього СРС.</p>
14	<p>«Обробка виключень». Класи опису виключних ситуацій. Власні виключні ситуації. <i>Exception</i>, <i>Error</i>, <i>Runtime Exception</i>.</p> <p>СРС: Створити власний клас для представлення та обробки власної виключної ситуації.</p>
15	<p>«Модульне тестування». Побудова та використання модульних тестів.</p> <p>СРС: Створити власний клас з методами для модульного тестування одного з класу написаного в рамках однієї з лабораторних робіт даного кредитного модуля.</p>
16	<p>«Управління пам'яттю, деструктори об'єктів та <i>Garbage Collector</i>». Система «збору сміття», критерій наявності посилань на об'єкти, деструктор.</p> <p>СРС: Ознайомитися з алгоритмами роботи системи «збору сміття» (<i>Garbage Collector</i>) в Java.</p>
17	<p>«Пакет <i>java.util.stream</i>». Функціональний стиль програмування. Чисті функції. Технологія <i>map-reduce</i>.</p> <p>СРС: Реалізувати алгоритм виборки з масиву за критерієм за допомогою пакету <i>java.util.stream</i>.</p>
18	<p>«Події та орієнтоване на події управління поведінкою програми». Події в Java. Модулі <i>AWT</i> та <i>Swing</i>. <i>Observer</i>, <i>JMS</i>.</p> <p>СРС: Реалізувати програму на мові програмування Java, яка використовує механізм подій.</p>

Заочна форма навчання

№ лекції	Назва теми лекції, перелік основних питань та завдання на СРС
1	<p>«Вступ в Java». Загальна характеристика мови Java. Порівняння з іншими мовами програмування. Прості типи та основи синтаксису Java.</p> <p>СРС: Частково або повністю виконати першу лабораторну роботу даного кредитного модуля на мові програмування Java.</p>
	<p>«Вступ в Java. Продовження». Класи та методи в Java. Пакети та інтерфейси.</p>

	<p>СРС: Створити декілька методів в декількох різних класах, які знаходяться в різних пакетах та реалізують різні інтерфейси.</p>
2	<p>«Робота з масивами в Java». Одновимірні, багатовимірні масиви та об'єкти, що представляють масиви в Java. СРС: Частково або повністю виконати другу лабораторну роботу даного кредитного модуля на мові програмування Java.</p> <p>«Робота зі строками в Java». Класи String, String Buffer, String Builder. СРС: Частково або повністю виконати третю лабораторну роботу даного кредитного модуля на мові програмування Java.</p> <p>«Методи класу. Перевантаження, статичність та доступ». Перевантаження методів. Перевантаження конструкторів. Управління доступом. Статичні елементи. СРС: Створити декілька перевантажених методів з різними модифікаторами доступу.</p> <p>«Властивості (поля) класу. Анонімні класи на прикладі сортування масиву об'єктів. Наслідування та управління видимістю». Властивості (поля). Наслідування властивостей. Управління областю видимості. Анонімні класи. Інтерфейс Comparator. Метод Compare. Сортування масиву об'єктів. Клас Arrays. Метод Sort. СРС: Створити клас з декількома полями з різними областями видимості. Створити клас, що наслідує попередній. Спробувати доступитися в ньому до полів попереднього класу та створити власні поля з такими ж іменами. Частково або повністю виконати четверту лабораторну роботу даного кредитного модуля на мові програмування Java.</p> <p>«Ієрархії об'єктів». Створення багаторівневих ієрархій об'єктів через наслідування та/або агрегацію. Клас Object. СРС: Частково або повністю виконати п'яту лабораторну роботу даного кредитного модуля на мові програмування Java.</p>
3	<p>«Наслідування методів. Поліморфізм». Наслідування методів. Поліморфізм. Специфікатори super та final. СРС: Частково або повністю виконати шосту лабораторну роботу даного кредитного модуля на мові програмування Java.</p> <p>«Реалізація поліморфізму». Реалізація поліморфізму. Перевизначення методів. Раннє та пізнє зв'язування. СРС: Створити клас з декількома методами. Створити клас, що наслідує попередній та перевизначає його методи. Створити екземпляр другого класу, посилання на який буде зберігати змінна першого класу. Спробувати доступитися через вказану змінну першого класу до перевизначеного методу з другого класу.</p> <p>«Колекції в Java». Колекції в Java. Зв'язані списки. Інтерфейси Set та List. Колекції в Java. Зв'язані списки. Інтерфейси Set та List. Методи toArray. Ітератори. Унікальність елементів у колекціях типу Set. Еквівалентність об'єктів. СРС: Спробувати створити власний клас, який реалізує будь-який інтерфейс колекції з реалізацією ітератора та еквівалентності.</p> <p>«Обробка виключень та модульне тестування». Класи опису виключних ситуацій. Власні виключні ситуації. Exception, Error, Runtime Exception. Побудова та використання модульних тестів. СРС: Створити власний клас для представлення та обробки власної виключної ситуації. Створити власний клас з методами для модульного тестування одного з класу написаного в рамках однієї з лабораторних робіт даного кредитного модуля.</p>

	<p>«Управління пам'яттю, деструктори об'єктів та Garbage Collector». Система «збору сміття», критерій наявності посилань на об'єкти, деструктор. СРС: Ознайомитися з алгоритмами роботи системи «збору сміття» (Garbage Collector) в Java.</p>
4	<p>«Пакет java.util.stream». Функціональний стиль програмування. Чисті функції. Технологія map-reduce. СРС: Реалізувати алгоритм виборки з масиву за критерієм за допомогою пакету java.util.stream.</p> <p>«Події та орієнтоване на події управління поведінкою програми». Події в Java. Модулі AWT та Swing. Observer, JMS. СРС: Реалізувати програму на мові програмування Java, яка використовує механізм подій.</p>

Лабораторні заняття

Очна форма навчання

№ з/п	Назва лабораторної роботи	Кількість ауд. годин
1	Основні типи та оператори мови програмування Java (Тема 1)	3
2	Масиви, які представлені у виді об'єктів (Тема 3)	3
3	Строки, які представлені у виді об'єктів (Тема 4)	3
4	Класи та їх властивості (Тема 6)	3
5	Відношення між класами, побудова ієрархій (Тема 7)	3
6	Наслідування та поліморфізм (Тема 8)	3
	Разом:	18

Заочна форма навчання

№ з/п	Назва лабораторної роботи	Кількість ауд. годин
1	Основні типи та оператори мови програмування Java (Тема 1)	2
2	Масиви, які представлені у виді об'єктів (Тема 3)	1
3	Строки, які представлені у виді об'єктів (Тема 4)	1
4	Класи та їх властивості (Тема 6)	1
5	Відношення між класами, побудова ієрархій (Тема 7)	2
6	Наслідування та поліморфізм (Тема 8)	1
	Разом:	8

6. Самостійна робота студента

Очна форма навчання

№ з/п	Назва теми, що виноситься на самостійне опрацювання	Кількість годин СРС
1	«Вступ в Java». Частково або повністю виконати першу лабораторну роботу даного кредитного модуля на мові програмування Java.	4

2	«Вступ в Java. Продовження». Створити декілька методів в декількох різних класах, які знаходяться в різних пакетах та реалізують різні інтерфейси.	4
3	«Робота з масивами в Java». Частково або повністю виконати другу лабораторну роботу даного кредитного модуля на мові програмування Java.	4
4	«Робота зі строками в Java». Частково або повністю виконати третю лабораторну роботу даного кредитного модуля на мові програмування Java.	4
5	«Методи класу. Перевантаження, статичність та доступ». Створити декілька перевантажених методів з різними модифікаторами доступу.	4
6	«Властивості (поля) класу. Анонімні класи на прикладі сортування масиву об'єктів». Частково або повністю виконати четверту лабораторну роботу даного кредитного модуля на мові програмування Java.	2
7	«Наслідування та управління видимістю». Створити клас з декількома полями з різними областями видимості. Створити клас, що наслідує попередній. Спробувати доступитися в ньому до полів попереднього класу та створити власні поля з такими ж іменами.	2
8	«Ієрархії об'єктів». Частково або повністю виконати п'яту лабораторну роботу даного кредитного модуля на мові програмування Java.	4
9	«Наслідування методів. Поліморфізм». Частково або повністю виконати шосту лабораторну роботу даного кредитного модуля на мові програмування Java.	4
10	«Реалізація поліморфізму». Створити клас з декількома методами. Створити клас, що наслідує попередній та перевизначає його методи. Створити екземпляр другого класу, посилання на який буде зберігати змінна першого класу. Спробувати доступитися через вказану змінну першого класу до перевизначеного методу з другого класу.	4
11	«Колекції в Java». Спробувати створити власний клас, який реалізує будь-який інтерфейс колекції (без реалізації ітератора та еквівалентності).	2
12	«Колекції в Java. Продовження». Реалізувати ітератор в рамках попереднього СРС.	1
13	«Колекції в Java. Продовження». Реалізувати перевірку еквівалентності окремих елементів та всієї колекції в рамках попереднього СРС.	1
14	«Обробка виключень». Створити власний клас для представлення та обробки власної виключної ситуації.	2
15	«Модульне тестування». Створити власний клас з методами для модульного тестування одного з класу написаного в рамках однієї з лабораторних робіт даного кредитного модуля.	2
16	«Управління пам'яттю, деструктори об'єктів та Garbage Collector». Ознайомитися з алгоритмами роботи системи «збору сміття» (Garbage Collector) в Java.	4
17	«Пакет java.util.stream». Реалізувати алгоритм виборки з масиву за критерієм за допомогою пакету java.util.stream.	5

18	«Події та орієнтоване на події управління поведінкою програми». Реалізувати програму на мові програмування Java, яка використовує механізм подій.	5
19	«Підготовка до МКР»	4
20	«Підготовка до заліку»	4
	Разом:	66

Заочна форма навчання

№ з/п	Назва теми, що виноситься на самостійне опрацювання	Кількість годин СРС
1	«Вступ в Java». Частково або повністю виконати першу лабораторну роботу даного кредитного модуля на мові програмування Java. Створити декілька методів в декількох різних класах, які знаходяться в різних пакетах та реалізують різні інтерфейси.	12
2	«Робота з масивами в Java». Частково або повністю виконати другу лабораторну роботу даного кредитного модуля на мові програмування Java.	6
3	«Робота зі строками в Java». Частково або повністю виконати третю лабораторну роботу даного кредитного модуля на мові програмування Java.	6
4	«Методи класу. Перевантаження, статичність та доступ». Створити декілька перевантажених методів з різними модифікаторами доступу.	6
5	«Властивості (поля) класу. Анонімні класи на прикладі сортування масиву об'єктів. Наслідування та управління видимістю». Створити клас з декількома полями з різними областями видимості. Створити клас, що наслідує попередній. Спробувати доступитися в ньому до полів попереднього класу та створити власні поля з такими ж іменами. Частково або повністю виконати четверту лабораторну роботу даного кредитного модуля на мові програмування Java.	6
6	«Ієрархії об'єктів». Частково або повністю виконати п'яту лабораторну роботу даного кредитного модуля на мові програмування Java.	6
7	«Наслідування методів. Поліморфізм». Частково або повністю виконати шосту лабораторну роботу даного кредитного модуля на мові програмування Java.	6
8	«Реалізація поліморфізму». Створити клас з декількома методами. Створити клас, що наслідує попередній та перевизначає його методи. Створити екземпляр другого класу, посилання на який буде зберігати змінна першого класу. Спробувати доступитися через вказану змінну першого класу до перевизначеного методу з другого класу.	7
9	«Колекції в Java». Спробувати створити власний клас, який реалізує будь-який інтерфейс колекції з реалізацією ітератора та еквівалентності.	7
10	«Обробка виключень та модульне тестування». Створити власний клас для представлення та обробки власної виключної ситуації. Створити власний клас з методами для модульного тестування одного з класу написаного в рамках однієї з лабораторних робіт даного кредитного модуля.	7
11	«Управління пам'яттю, деструктори об'єктів та Garbage Collector».	7

	<i>Ознайомитися з алгоритмами роботи системи «збору сміття» (Garbage Collector) в Java.</i>	
12	«Пакет java.util.stream». Реалізувати алгоритм виборки з масиву за критерієм за допомогою пакету java.util.stream.	7
13	«Події та орієнтоване на події управління поведінкою програми». Реалізувати програму на мові програмування Java, яка використовує механізм подій.	7
14	«Підготовка до МКР»	7
15	«Підготовка до заліку»	7
	Разом:	104

Політика та контроль

7. Політика навчальної дисципліни (освітнього компонента)

- правила захисту лабораторних робіт:
 - студент надає програмний код до лабораторної роботи,
 - студент захищає наданий програмний код шляхом співбесіди;
- політика дедлайнів та перескладань:
 - студент має право здавати та перездавати лабораторні роботи до дня заліку включно,
 - обмеження на кількість перескладань відсутнє;
- політика щодо академічної доброчесності:
 - студент має право захищати програмний код, який був написаний не лише ним, але в цьому випадку студент має розуміти «що» та «як» виконує цей код, вміти використати його як основу для внесення модифікацій, які можуть бути запропоновані викладачем.

8. Види контролю та рейтингова система оцінювання результатів навчання (PCO)

Види контролю з навчальної дисципліни «Основи розробки програмного забезпечення на платформі Java» включають:

Лабораторні роботи:

Заплановано самостійне виконання шести лабораторних робіт. Темі лабораторних робіт узгоджені у часі та за змістом з темами лекцій. Виконання лабораторних робіт у повному обсязі дозволяє набутти практичних навичок програмування на платформі Java.

Поточний контроль:

Передбачено проведення контрольної роботи (МКР)

Семестровий контроль:

Залік проводиться у вигляді співбесіди зі студентом для об'єктивного визначення рівня знань, умінь та практичних навичок, отриманих за семестр

Семестровий рейтинг студента складається з балів, які він отримує за види робіт відповідно до таблиці 1.

Оцінювання окремих видів навчальної роботи студента (у балах)

Вид навчальної роботи	Всього за видом роботи
Виконання та захист лабораторної роботи № 1	15
Виконання та захист лабораторної роботи № 2	15
Виконання та захист лабораторної роботи № 3	15
Виконання та захист лабораторної роботи № 4	15
Виконання та захист лабораторної роботи № 5	15
Виконання та захист лабораторної роботи № 6	15
МКР	Rк 10
	Rп 100
Залік (Rз) за бажанням	20

Індивідуальний поточний рейтинг студента (**Rп**) складається з балів, які він отримує за виконання лабораторних робіт (**Rл**) і МКР (**Rк**). Протягом семестру студенти виконують 6 лабораторних робіт. Максимальна кількість балів за кожну лабораторну роботу – 15. Бали нараховуються за:

- теоретична складова – 7 балів,
- практична складова – 8 балів.

Максимальний можливий бал за лабораторну роботу – 15 балів. Максимальна кількість балів за всі лабораторні роботи $6 \times 15 = 90$ балів.

Розрахунок розміру шкали (R) рейтингу.

Сума вагових балів контрольних заходів протягом семестру становить:

$Rп = Rл + Rк$, де **Rп** – семестровий рейтинг студента (МКР, лабораторні роботи).

Необхідною умовою допуску студента до заліку є його індивідуальний семестровий рейтинг (**Rп**), не менший, ніж 40 балів, та відсутність заборгованості з лабораторних робіт та МКР. При невиконанні згаданих вимог студент до заліку не допускається.

Таблиця відповідності рейтингових балів оцінкам за університетською шкалою:

Кількість балів	Оцінка
100-95	Відмінно
94-85	Дуже добре
84-75	Добре
74-65	Задовільно
64-60	Достатньо
Менше 60	Незадовільно
Не виконані умови допуску	Не допущено

9. Додаткова інформація з дисципліни (освітнього компонента)

В рамках вивчення дисципліни «Основи розробки програмного забезпечення на платформі Java» допускається зарахування балів, одержаних в результаті дистанційних курсів за умови погодження програми даного курсу з викладачем.

Робочу програму навчальної дисципліни (силабус):

Складено старшим викладачем кафедри обчислювальної техніки *Алещенко Олексієм Вадимовичем*

Ухвалено кафедрою обчислювальної техніки (протокол № 10 від 25.05.2022)

Погоджено Методичною комісією факультету (протокол № 10 від 09.06.2022)