

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Одеський державний екологічний університет

ЗАТВЕРДЖЕНО

на засіданні групи забезпечення спеціальності

від « 10 » 11 2022 року

протокол № 9

Голова групи к.геогр.н., доцент Кузніченко С.Д.

УЗГОДЖЕНО

Т.в.о декан факультету комп'ютерних наук,

управління та адміністрування

доктор філософії з КН, Бучинська І.В.

СИЛЛАБУС

навчальної дисципліни

Крос-платформне програмування

(назва навчальної дисципліни)

122 – Комп'ютерні науки

(шифр та назва спеціальності)

Комп'ютерні науки

(назва освітньої програми)

бакалавр

(рівень вищої освіти)

денна/заочна

(форма навчання)

4 / 5

(рік навчання)

7 (д/ф)

(семестр навчання)

6 /180

(кількість кредитів ЄКТС/годин)

залік

(форма контролю)

Інформаційних технологій

(кафедра)

Одеса, 2022 р.

Автор: Кузніченко С.Д., к.геогр.н., доцент

Клепатська В.В., асистент

Поточна редакція розглянута на засіданні кафедри інформаційних технологій від
18 08 20 22 року, протокол № 1 .

Викладачі: Лекційний модуль – Кузніченко Світлана Дмитрівна, к.геогр.н., доцент

Цира Олександра Василівна, к.ф.н., доцент

Практичний модуль (лабораторні роботи) – Клепатська Вікторія Вікторівна

Перелік попередніх редакцій

Прізвища та ініціали авторів	Дата, № протоколу	Дата набуття чинності

1. ОПИС НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Мета	Теоретичне вивчення та отримання практичних навичок розробки крос-платформних програмних систем та використання засобів крос-платформного програмування.
Компетентність	Знання архітектури та стандартів компонентних моделей, здатність вирішувати проблеми взаємодії різних програмних платформ.
Результат навчання	Вирішувати проблеми взаємодії різних програмних платформ в розподілених корпоративних інформаційних системах.
Базові знання	<ol style="list-style-type: none"> 1. Базові принципи створення крос-платформних програмних систем; 2. Класифікація крос-платформних мов програмування; 3. Середовища розробки крос-платформного програмного забезпечення; 4. Бібліотеки для створення платформи-незалежного програмного забезпечення.
Базові вміння	<ol style="list-style-type: none"> 1. Проектувати компоненти програмного забезпечення; 2. Проектувати крос-платформного графічного інтерфейсу користувача; 3. Реалізовувати прототипи архітектури програмного забезпечення; 4. Інтегрувати компоненти в систему; 5. Установлювати, налаштовувати та обслуговувати системне, інструментальне і прикладне програмне забезпечення та інформаційні системи.
Базові навички	Програмування на крос-платформній мові програмування Java.
Пов'язані силабуси	—
Попередня дисципліна	Об'єктно-орієнтоване програмування
Наступна дисципліна	—
Кількість годин	<div style="text-align: right;">денна / заочна</div> <div>Лекційний модуль: 30 год. / 2 год.;</div> <div>Практичний модуль (лабораторні роботи): 30 год. / 4 год.;</div> <div>Самостійна робота студентів: 120 год. / 166 год.</div> <div>Консультації: — / 8 год.</div>

ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

1.1 Лекційний модуль

Код	Назва модуля та тем	Кількість годин			
		денна		заочна	
		аудиторні	СРС	аудиторні	СРС
ЗМ-Л1	<i>Розробка додатків з GUI мовою Java</i>	1			2
	• Вступ. Предмет, мета і задачі дисципліни	6	2		12
	• Компонентна ідеологія	6	4+5		17
	• Багатопотокове програмування. Потоки введення/виведення (I/O Streams)				
ЗМ-Л2	<i>Архітектура та проектування компонентних систем</i>	4	2+5		15
	• Розробка та збирання компонентів, розподілена архітектура компонентних систем, компонентно-орієнтоване проектування	4	4		14
	• Колекції в Java	9	6		18
	• Розробка мережевих програм				
ЗМ-І3	Індивідуальне завдання (реферат)		10		10
	Настановні лекції			2	
Разом:		30	38	2	88

Консультації:

Цира Олександра Василівна, **понеділок 12⁴⁵ – 14²⁰ год.** 320 ауд. НЛК№1
Додаткова контактна інформація: aleksandra.tsyra@gmail.com.

1.1 Практичний модуль (лабораторні роботи)

Код	Назва модуля та тем	Кількість годин			
		денна		заочна	
		аудиторні	СРС	аудиторні	СРС
ЗМ-П1	1) Програмування аплетів з елементами мультимедіа для Web сторінок Internet	4	12		12
	2) Використання потоків і створення анімації в Java	6	12	2	16
	3) Створення графічних застосувань. Використання системи Swing та JavaFX	6	12		12

ЗМ-П2	1) Програмування потоків введення/виведення	6	12	2	12
	2) Програмування колекцій в Java	4	12		12
	3) Використання сокетів у розподілених застосунках	4	12		12
Разом:		30	72	4	76

Перелік лабораторій та лабораторного обладнання:

Лабораторні роботи проводяться в комп'ютерних класах кафедри інформаційних технологій (ауд. 319, 324, 327, 328, 329). Під час проведення лабораторних робіт використовується наступне програмне забезпечення: ПК з ОС Windows інструментальні засоби jdk java 2.x, інтегральне середовище розробника Eclipse 3.x.

Консультації:

Клепатська Вікторія Вікторівна, вівторок 10⁴⁵ – 12²⁰ год. 330 ауд. НЛК №1
Додаткова контактна інформація: victoria.klepatska@gmail.com.

1.1 Самостійна робота студента та контрольні заходи (денна форма)

Код	Завдання на СРС та контрольні заходи	Кількість годин	Строк проведення
ЗМ-Л1	• Підготовка до лекційних занять	8	1-7 тижні
	• Підготовка до модульної контрольної роботи № 1	6	1-7 тижні
	• Модульна контрольна робота № 1 (обов'язкова)	2	7 тиждень
ЗМ-Л2	• Підготовка до лекційних занять	10	8-15 тижні
	• Підготовка до модульної контрольної роботи № 2	10	8-15 тижні
	• Модульна контрольна робота № 2 (обов'язкова)	2	15 тиждень
ЗМ-П1	• підготовка до усного опитування напередодні відповідної лабораторної роботи (обов'язкове)	12	1-8 тижні
	підготовка до захисту звіту з лабораторних робіт (обов'язковий)	19	1-8 тижні
ЗМ-П2	• підготовка до усного опитування напередодні відповідної лабораторної роботи (обов'язкове)	12	9-15 тижні
	підготовка до захисту звіту з лабораторних робіт (обов'язковий)	19	9-15 тижні
ЗМ-ІЗ	Підготовка до захисту індивідуального завдання (реферат)	10	15 тиждень
	• Підготовка до залікової контрольної роботи	10	15 тиждень
Разом:		120	

1.1 Самостійна робота студента та контрольні заходи (заочна форма)

Код	Завдання на СРС та контрольні заходи	Кількість годин	Строк проведення
ЗМ-ЛІ	• Вивчення тем лекційного модуля	34	вересень-листопад
	• Модульна контрольна робота № 1 (обов'язкова)	2	листопад
ЗМ-ЛІІ	• Вивчення тем лекційного модуля	40	грудень-лютий
	• Модульна контрольна робота № 2 (обов'язкова)	2	лютий
ЗМ-ЛІІІ	• Підготовка до виконання лабораторних робіт • Звіт про виконання лабораторних робіт №1 та 3 (обов'язковий)	26	вересень-листопад
	• Підготовка до усного опитування до лабораторної роботи №2 (обов'язковий)	14	сесія
ЗМ-ЛІІІІ	• Підготовка до виконання лабораторних робіт • Звіт про виконання лабораторних робіт №5 та 6 (обов'язковий)	24	грудень-лютий
	• Підготовка до усного опитування до лабораторної роботи №4 (обов'язковий)	12	сесія
ЗМ-ЛІІІІІ	Підготовка індивідуального завдання (реферат)	10	сесія
	• Підготовка до залікової контрольної роботи	10	сесія
Разом:		174	

Електронний курс з дисципліни доступний за посиланням: <http://dpt15s.odeku.edu.ua/course/view.php?id=30>.

1. Методика проведення та оцінювання контрольного заходу для ЗМ-ЛІ

Контроль проводиться після вивчення лекційного матеріалу модуля ЗМ-ЛІ в формі письмової модульної контрольної роботи МКР-1 в якій студенти відповідають на 7 запитань. Результати роботи оформлюються на окремому аркуші. Час, що виділяється на виконання МКР-1 визначається при видачі завдання і не перевищує 1 академічної години.

Критерії оцінювання результатів контрольного заходу: максимальна оцінка за контрольну роботу складає 20 балів; розподіл балів за правильні відповіді має наступний вигляд:

Номер запитання	Кількість балів за правильну відповідь
1	2
2	3
3	2
4	2
5	4
6	3
7	4
Разом:	20

Критерії оцінювання результатів контрольного заходу: відмінно (18-20 балів); добре (14,8-17,8 балів); задовільно (12-14,79 балів); незадовільно (менше 12 балів).

2. Методика проведення та оцінювання контрольного заходу для ЗМ-Л2

Контроль проводиться після вивчення лекційного матеріалу модуля ЗМ-Л2 в формі письмової модульної контрольної роботи МКР-2 в якій студенти відповідають на 6 запитань. Результати роботи оформлюються на окремому аркуші. Час, що виділяється на виконання МКР-2 визначається при видачі завдання і не перевищує 1 академічної години.

Максимальна оцінка за контрольну роботу складає 20 балів. Розподіл балів за правильні відповіді має наступний вигляд:

Номер запитання	Кількість балів за правильну відповідь
1	2
2	4
3	3
4	3
5	4
6	4
Разом:	20

Критерії оцінювання результатів контрольного заходу: відмінно (18-20 балів); добре (14,8-17,8 балів); задовільно (12-14,79 балів); незадовільно (менше 12 балів).

У разі дистанційного навчання, проведення МКР-1 та МКР-2 буде проходити в системі е-навчання — **moodle** (<http://dpt15s.odku.edu.ua/course/view.php?id=30>).

3. Методика проведення та оцінювання контрольного заходу для ЗМ-ІЗ.

Індивідуальне завдання (ІЗ) виконується з метою систематизації закріплення, поглиблення та узагальнення знань, одержаних студентом за час навчання та придбання практичних навичок у вигляді написання реферату.

У процесі виконання ІЗ студент повинен застосувати набуті теоретичні знанням і практичні навички та продемонструвати здібності до науково-дослідної роботи і вміння творчо мислити, навчитися вирішувати науково-прикладні актуальні задачі.

Текст роботи виконується на аркушах формату А4. Заповнення аркушів виконується чорним кольором на одній сторінці білого паперу, шрифтом типу Times New Roman розміру 14 pt. Для забезпечення вимоги про кількість рядків на сторінці рекомендується встановлювати міжрядковий інтервал таким, що дорівнює 1,0. Загальний обсяг роботи повинен бути 12-15 сторінок. Структура роботи: титульний аркуш, зміст, теорія, перелік посилань та **документ про**

оригінальність документу (звіт з ресурсу/антиплагіт, яким перевірявся реферат на унікальність).

Максимальна оцінка за ІЗ складає 10 балів. Критерії оцінювання результатів контрольного заходу: правильна відповідь на 4 і більше запитань – відмінно (9-10 балів), правильна відповідь на 2-3 запитань – добре (7,4-8,9 балів), правильна відповідь на 1 запитання – задовільно (6-7,3 балів), жодної правильної відповіді (ні на одне питання) – незадовільно (менше 6 балів).

4. Методика підсумкового оцінювання контрольних заходів для всіх лекційних модулів

Підсумкова оцінка за всі лекційні модулі дорівнює сумі набраних балів за лекційні модулі ЗМ-Л1, ЗМ-Л2 та ЗМ-ІЗ яка не може перевищувати 50 балів. Якщо обов'язкові заходи контролю виконуються студентом після строків, визначених у програмі навчальної дисципліни, кількість балів, що може отримати студент, не може перевищувати 60% від максимально можливої для цієї форми контролю.

5. Методика проведення та оцінювання контрольного заходу для ЗМ-П1 і ЗМ-П2 (денна форма)

За кожний практичний модуль встановлена максимальна оцінка 25 балів. Кожний практичний модуль складається з трьох лабораторних робіт, за двома з яких встановлена максимальна оцінка 8 балів, а за третьої – 9 балів.

Контроль по кожній лабораторній роботі проводиться в формі:

- *усного опитування* при підготовці до кожної лабораторної роботи з метою допуску до її виконання (кількість запитань – до 3, максимальна кількість балів – 3),
- *захисту результатів* лабораторної роботи наведених у звіті до лабораторної роботи (кількість запитань залежить від ходу виконання студентом роботи і якості звіту, максимальна кількість балів за першу і другу лабораторну роботу – 5; за третю – 6 балів).

Для кожної лабораторної роботи, якщо студент за *усне опитування* одержав 1 і менше балів він не допускається до виконання роботи, а якщо більше – допускається.

Підсумковою оцінкою за кожну лабораторну роботу буде сума балів за *усне опитування* і *захист результатів*. Якщо обов'язкові заходи контролю виконуються студентом після строків, визначених у програмі навчальної дисципліни, кількість балів, що може отримати студент, не може перевищувати 60% від максимально можливої для цієї форми контролю.

Підсумковою оцінкою за кожний практичний модуль буде сума балів за всі лабораторні роботи. Критерії оцінювання результатів контрольного заходу для ЗМ-П1 та ЗМ-П2: 22 бали і більше – відмінно, 19-21,9 – добре, 15-18,9 балів – задовільно, менше 15 балів – незадовільно.

1. Методика проведення та оцінювання контрольного заходу для ЗМ-ПІІ ЗМ-ПІІ (заочна форма)

За кожний практичний модуль встановлена максимальна оцінка 25 балів. Кожний практичний модуль складається з трьох лабораторних робіт, за двома (1, 3, 5 та 6 лабораторні роботи) з яких встановлена максимальна оцінка 8 балів, а за третьої (2 та 4 лабораторні роботи) – 9 балів.

Контроль по 1, 3, 5 та 6 лабораторним роботам проводиться в формі:

- перевірки звіту з лабораторної роботи.

Контроль по 2 та 4 лабораторним роботам проводиться в формі:

- усного опитування при підготовці до лабораторної роботи з метою допуску до її виконання (кількість запитань – до 4, максимальна кількість балів – 4),
- захисту *результатів* лабораторної роботи наведених у звіті до лабораторної роботи (кількість запитань залежить від ходу виконання студентом роботи і якості звіту, максимальна кількість балів – 5).

Для 2 та 4 лабораторної роботи, якщо студент за усне опитування одержав 1 і менше балів він не допускається до виконання роботи, а якщо більше – допускається.

Для лабораторної роботи №2 та №4 при *захисті результатів* студент може одержати від 1 до 5 балів.

Підсумковою оцінкою за кожну лабораторну роботу буде сума балів за *усне опитування і захист результатів*.

Підсумковою оцінкою за кожний практичний модуль буде сума балів за всі лабораторні роботи. Критерії оцінювання результатів контрольного заходу для ЗМ-ПІІ та ЗМ-ПІІ: 22 бали і більше – відмінно, 19-21,9 – добре, 15-18,9 балів – задовільно, менше 15 балів – незадовільно.

2. Методика оцінювання за всіма змістовними модулями

Підсумковою оцінкою за всіма змістовними модулями (ОЗ) буде сума балів за лекційні модулі, практичні модулі та за виконання індивідуального завдання (реферату).

3. Методика проведення та оцінювання підсумкового контрольного заходу

Підсумковий контрольний захід проводиться у формі залікової контрольної роботи (ЗКР) тестового типу в якій студенти відповідають на 20 запитань. Результати роботи оформлюються на окремому аркуші. Час, що виділяється на виконання залікової контрольної роботи визначається при видачі завдання і не перевищує 1 академічної години.

Студент вважається допущеним до підсумкового контролю з навчальної дисципліни, якщо він виконав всі види робіт поточного контролю, передбачені

програмою дисципліни і набрав за накопичувальною системою суму балів не менше 50% від максимально можливої за теоретичну (більше 25 балів) та практичну (більше 25 балів) частини дисципліни.

Максимальна оцінка за залікову контрольну роботу (ОЗКР) складає 100 балів. Оцінка еквівалентна відсотку правильних відповідей на запитання. Критерії оцінювання результатів залікової контрольної роботи: 90 балів і більше правильних відповідей – відмінно, 74...89,9 балів – добре, 60...73,9 балів – задовільно, менше 60 балів – незадовільно.

4. Методика підсумкового оцінювання за дисципліну

Сума балів, яку одержав студент за лекційні модулі, за практичний модуль і за залікову контрольну роботу формують інтегральну оцінку студента з навчальної дисципліни. Інтегральна оцінка (В) за дисципліну розраховується за формулою:

$$B = 0,75 \times OЗ + 0,25 \times OЗКР,$$

де ОЗ – кількісна оцінка (у балах від максимально можливої в 100 балів) за всіма змістовними модулями, ОЗКР – кількісна оцінка (у балах від максимально можливої в 100 балів) залікової контрольної роботи.

Інтегральна оцінка (В) за дисципліну за всіма системами оцінювання наведена у наступній таблиці:

Визначення	За системою університету (у відсотках)	За національною системою	За шкалою ECTS
відмінне виконання лише з незначною кількістю помилок	90 – 100	зараховано	A
вище середнього рівня з кількома помилками	82 – 89,9	зараховано	B
В загальному правильна робота з певною кількістю грубих помилок	74 – 81,9	зараховано	C
непогано, але зі значною кількістю помилок	64 – 73,9	зараховано	D
виконання задовольняє мінімальні критерії	60 – 63,9	зараховано	E
з можливістю перескладання	35 – 59,9	не зараховано	FX
з обов'язковим повторним курсом навчання	1 – 34,9	не зараховано	F

При цьому позитивна інтегральна оцінка з дисципліни (зараховано) одержується студентом за наступних умов:

- студент не має наприкінці семестру заборгованості з дисципліни,
- студент має на останній день семестру підсумкову суму балів поточного контролю достатню для одержання позитивної оцінки ($OZ \geq 60\%$),
- студент має $OZKP \geq 50\%$ від максимально можливої суми балів за залікову контрольну роботу.

3. РЕКОМЕНДАЦІЇ ДО САМОСТІЙНОЇ РОБОТИ СТУДЕНТІВ

Рекомендується наступний порядок вивчення дисципліни «Крос-платформне програмування»:

–зміст кожної теми курсу вивчається за допомогою навчальної та методичної літератури, що наведена в списку;

–після засвоєння змісту кожної теми курсу потрібно відповісти на «запитання самоперевірки», що наведені у силабусу і відповідній літературі;

–якщо виникли питання при вивченні теоретичного матеріалу або при виконанні контрольних робіт, то потрібно звернутись до викладача, який проводив лекції.

3.1. Модуль ЗМ-Л1 «Розробка додатків з GUI мовою Java»

3.1.1. Повчання

Розділи модуля ЗМ-Л1 формують у студентів уявлення про загальні принципи компонентної ідеології: про властивості компонентів, специфікації інтерфейсу як контракту, модель посилань. Розглядається особливості використання даних принципів при створенні багатопотокових застосувань мовою Java.

При вивченні цих розділів необхідно звернути увагу на концепції потоків виконання (threads) і введення/виведення (streams), а також модель обробки подій в Java.

Програма ЗМ-Л1. Розробка додатків з GUI мовою Java

Компонентна ідеологія [4, с.7-19]

Визначення та властивості компонентів [2, с.64-81]

Специфікації інтерфейсу як контракту [2, с.64-81]

Модель посилань [2, с.64-81; 3, с.275-287]

Стратегії інтеграції програмного забезпечення [4, с.7-21]

Багатопотокове програмування [1, с.4-40; 3, с.328-350]

Використання потоків і створення анімації в Java [1, с.9-14; 3, с.88-97]

Синхронізація потоків [1, с.14-32; 3, с.328-350]

Переривання потоків [1, с.32-39; 3, с.328-350]

Програмування потоків введення/виведення [1, с.44-70; 3, с.211-217]

Шаблони Adapter і Decorator [3, с.220-229]

3.1.2. Питання для самоперевірки

Запитання, що входять до тестів до модуля ЗМ-Л1 і являють собою необхідний мінімум знань, який потрібний для засвоєння дисципліни «Крос-платформне програмування», наведені нижче:

- 1) *Дайте визначення програмного компонента. Що таке інтерфейс ний контракт?**
- 2) Дайте порівняльну характеристику компонентним моделям COM, JavaBeans, Corba.
- 3) Назвіть основні елементи компонентної моделі.
- 4) *Які відмінності компонентів від класів?**
- 5) *Які два способи використовуються для реалізації потоків в Java?**
- 6) Як виконується зупинка або завершення потоку?
- 7) Які операції над потоками визначені в Java?
- 8) *Як виконується переривання потоку і перевірка стану переривання?**
- 9) *Що таке потоки-демони і чим вони відрізняються від звичайних потоків?**
- 10) *Як задати і отримати значення пріоритету для потоку?**
- 11) Як можна створити групу потоків і які операції визначені для групи потоків в Java?
- 12) Які засоби синхронізація потоків є в Java?
- 13) *Як виконується синхронізація потоків за допомогою оператора synchronized?**
- 14) *Як виконується синхронізація потоків за допомогою методів wait() і notify()?**
- 15) Чим відрізняється робота методу wait() з параметром і без параметра?
- 16) *Чим відрізняються методи Thread.sleep() і Object.wait()?**
- 17) *Як працює метод Thread.join()?**
- 18) *Що таке dead lock?**
- 19) *Як правильно завершити роботу потоку? **

* - Курсивом виділені питання для самоперевірки базових результатів навчання -знань, вмінь, навичок

- 20) На якому об'єкті відбувається синхронізація при виклику static synchronized методу?
- 21) Для чого використовується ключове слово volatile? *
- 22) Яким буде результат виконання наступного програмного кода?
- a)

```
public static void main(String[] args)
    throws InterruptedException{
    Object obj = new Object();
    synchronized (obj) {
        obj.wait();
    }
}
```
- b)

```
public static void main(String[] args)
    throws InterruptedException{
    Object obj0 = new Object();
    Object obj = obj0;
    synchronized (obj) {
        obj0.wait();
    }
}
```
- c)

```
public static void main(String[] args)
    throws InterruptedException{
    synchronized (new Object) {
        new Object.wait();
    }
}
```
- 1) Для яких цілей використовується клас File? Які операції над файлами визначені в класі File?*
- 2) Як в класі File реалізована фільтрація файлів?
- 3) Які засоби для навігації по файлам і каталогам забезпечує клас JFileChooser?*
- 4) Що таке потік даних? Які види потоків визначені в Java?*
- 5) Які методи читання і запису даних визначені в класах InputStream і OutputStream?
- 6) Як в Java реалізований байтовий обмін даними з файлами?*
- 7) Для яких цілей використовуються класи ByteArrayInputStream і ByteArrayOutputStream?
- 8) Як в Java виконується об'єднання двох потоків вводу в один потік?
- 9) Які засоби буферизації потоків введення-виведення визначені в Java?

* - Курсивом виділені питання для самоперевірки базових результатів навчання - знань, вмінь, навичок

- 10) Як в Java виконується виведення байтових і символьних потоків на друк?
- 11) *Які методи читання і запису даних визначені в класах Reader і Writer?**
- 12) Як в Java реалізований перехід між байтовими і символьними потоками?
- 13) Як в Java реалізований символьний обмін даними з файлами?
- 14) *Як в Java реалізований введення-виведення в символьні масиви?**
- 15) *Для яких цілей використовується клас StreamTokenizer? **
- 16) Які методи аналізу даних визначено в класі StreamTokenizer?

3.1. Модуль ЗМ-Л2 «Архітектура та проектування компонентних систем»

3.1.1. Повчання

Розділи модуля ЗМ-Л2 формують у студентів уявлення про архітектуру та проектування компонентних систем, а саме: розробку та збирання компонентів, ідеологію компонентно-орієнтованого проектування.

При вивченні цих розділів необхідно звернути увагу на програмування колекцій, розробку мережевих програм та використання сокетів у розподілених застосунках.

Програма ЗМ-Л2. Архітектура та проектування компонентних систем

Архітектура та проектування компонентних систем [4, с.210-224]

Розробка та збирання компонентів [2, ст.244-276; 4, с.228-236]

Розподілена архітектура компонентних систем [4, с.214-217]

Компонентно-орієнтоване проектування [2, с.226-242; 4, с.228-236]

Бібліотека колекцій в Java [1, с.59-60; 3, с.205-210]

Програмування колекцій [1, с.59-71]

Розробка мережевих програм [1, с.73-84; 3, с.350-355]

Використання сокетів у розподілених додатках [1, с.77-80; 3, с.356-359]

Датаграми і протокол UDP [1, с.83-85; 3, с.360-364]

3.1.2. Питання для самоперевірки

Запитання, що входять до тестів до модуля ЗМ-Л2 і являють собою необхідний мінімум знань, який потрібний для засвоєння дисципліни «Крос-

* - Курсивом виділені питання для самоперевірки базових результатів навчання -знань, вмінь, навичок

платформне програмування», наведені нижче:

- 1) *Як визначаються колекції в мові Java? Які компоненти містить схема колекцій в Java? **
- 2) *Які інтерфейси колекцій визначені в мові Java? **
- 3) *Які операції над колекціями визначені за допомогою методів інтерфейсу Collection? **
- 4) *Які методи для перегляду елементів колекцій визначені в інтерфейсі Iterator? **
- 5) *Які додаткові методи для колекцій визначені в інтерфейсі List?*
- 6) *Які методи перегляду списку містить інтерфейс ListIterator?*
- 7) *Як упорядковано об'єкти з використанням інтерфейсу Comparable в програмах на мові Java?*
- 8) *Які методи (додатково до методів інтерфейсу Set) реалізує інтерфейс SortedSet? **
- 9) *Як визначається інтерфейс Map, і які операції визначені в цьому інтерфейсі? **
- 10) *Які методи (додатково до методів інтерфейсу Map) реалізує інтерфейс SortedMap? **
- 11) *Які типи реалізацій колекцій існують у мові Java? Дайте коротку характеристику кожного типу.*
- 12) *Які абстрактні класи реалізації колекцій визначені в Java? Дайте коротку характеристику кожного класу.*
- 13) *Як за допомогою класу ArrayList реалізуються масиви із змінними розмірами в програмах на мові Java? **
- 14) *Як в класі LinkedList реалізуються пов'язані списки?*
- 15) *Як в класі HashMap в Java реалізуються відображення з підтримкою хеш-таблиць?*
- 16) *Які операції над колекціями забезпечують методи класу Collections?*
- 17) *Для чого призначений і яке значення повертає метод getAllByName() класу java.net.InetAddress? **
- 18) *Поясніть, в чому полягає відмінність між методами getByName(null) і getLocalHost()? **
- 19) *Як отримати вміст сторінки, використовуючи її URL?*
- 20) *Які дії необхідно зробити для встановлення TCP з'єднання між двома java-додатками?*
- 21) *Які дії необхідно зробити для обміну даними по UDP протоколу? **

* - Курсивом виділені питання для самоперевірки базових результатів навчання - знань, вмінь, навичок

3.1. Модуль ЗМ-П1 „Практичний модуль”

При вивченні практичного модуля студенти набувають уміння програмування аплетів та графічних додатків з використанням системи Swing, роботи з потоками і створення анімації в Java [21,22].

При вивченні цього модуля необхідно звернути увагу на практичне застосування одержаних теоретичних знань про концепцію організації потокової моделі, а також моделі делегування подій в Java.

Перевірка якості засвоєних знань і одержаних навичок при вивченні цього модуля здійснюється викладачем під час проведення лабораторних занять шляхом усного опитування з наведених для теоретичного модуля ЗМ-Л1 питань і перевіркою якості виконання лабораторної роботи.

3.2. Модуль ЗМ-П2 „Практичний модуль”

При вивченні практичного модуля студенти набувають уміння використання потоків вводу/виводу та колекцій в програмах на мові Java, а також створення мережових додатків з використанням бібліотеки java.net. [21,22].

При вивченні цього модуля необхідно звернути увагу на практичне застосування одержаних теоретичних знань про організацію взаємодії з файловою системою комп'ютера, потоків вводу/виводу та управління мережевими з'єднаннями на рівні сокетів і дейтаграм.

Перевірка якості засвоєних знань і одержаних навичок при вивченні цього модуля здійснюється викладачем під час проведення лабораторних занять шляхом усного опитування з наведених для теоретичного модуля ЗМ-Л2 питань і перевіркою якості виконання лабораторної роботи.

3.5 Модуль ЗМ-ІЗ

Виконання індивідуального завдання ЗМ-ІЗ полягає в написанні та захисту реферату за відповідною темою. Перед допуском до захисту електронна версія наданих текстових документів обов'язково повинна пройти перевірку на оригінальність. Документ повинен мати оригінальність не менш ніж 40 %. Документ про оригінальність додається до роботи у вигляді звіту.

3.5.1 Перелік тем для виконання індивідуального завдання ЗМ-ІЗ.

1. Історія розвитку технологій Java EE.
2. Встановлення з'єднання за протоколом TCP.
3. Взаємодія за протоколом UDP.
4. Підходи до розроблення мережних багатопоточних додатків.
5. Переваги використання селекторів у мережних додатках.
6. Асинхронна робота мережних додатків.
7. Використання сокетів Берклі.

8. Інтерфейс передачі повідомлень (MPI).
9. Реалізація протоколів прикладного рівня.
10. Підходи до реалізації Proxy-сервера.
11. Підходи до реалізації SendMail-сервера.
12. Роль RMI в архітектурі Java EE.
13. Обробка запитів сервлетами з використанням захищеного протоколу HTTPS.
14. Механізми відстеження сесій в розподіленому Web-додатку.
15. Механізми програмної підтримки безпеки в сервлетах та JSP.
16. Шифрування, архівування та кешування даних в сервлетах.
17. Управління станом в Web-додатку на базі сервлетів та JSP.
18. Використання JSP 2.0 Expression language.
19. Підходи до реалізації простого Web-сервера.
20. Анотації та "ін'єкція залежностей" (Dependency Injection) в EJB 3.0.
21. Управління транзакціями в EJB 3.0.
22. Використання EJB 3.0 Timer Services в додатках Java EE.
23. Використання сервісів Java Authentication і Authorization.
24. Клієнти сесійних компонентів: локальний, видалений та Web-служба.
25. Переваги сервіс-орієнтованої архітектури інформаційної системи підприємства.
26. Сучасні підходи до створення ESB.
27. Програмне забезпечення за вимогою.
28. Робота з сервісами Google.
29. Робота з сервісами Amazon.
30. Програмне забезпечення як послуга.
31. Порівняння JWS-RS і JXA-WS Web-служб.
32. Використання RIA-додатків в бізнесі.
33. Використання RIA-додатків в іграх.
34. Використання RIA-додатків у навчальному процесі.
35. Продуктивність 2D графіки в RIA-додатках.
36. Продуктивність 3D графіки в RIA-додатках.
37. Спеціалізовані потоки в JavaFX.
38. Використання потокового відео та аудіо в додатках JavaFX.
39. Ефективність використання об'єктно-реляційного відображення в програмному забезпеченні рівня підприємства.
40. Відміна Hibernate від інших фреймворків об'єктно-реляційного відображення.
41. Інтеграція Hibernate з технологіями Java EE.
42. Реалізація зв'язку між таблицями в Hibernate.
43. Підключення Hibernate до серверів ApacheTomcat та GalassFish.
44. Hibernate Envers: аудіювання записів.
45. Критерії користувача в Hibernate.
46. Типи користувача в Hibernate.
47. Використання Spring в розподілених додатках.
48. Web-патерн MVC фреймворку Spring.
49. Можливості посередницького рівня Data Access у фреймворку Spring.

3.5.2 Посилання на літературу.

1. В. Л. Бурячок, Р. В. Киричок, П. М. Складанний. Основи інформаційної та кібернетичної безпеки : Навчальний посібник. К. , 2018. 320 с.
2. Б. Ю. Жураківський, І. О. Зенів. Розробка та реалізація мережних протоколів: Навчальний посібник [Електронний ресурс], 2020. 462 с.
3. Л.М. Олещенко. Організація комп'ютерних мереж: конспект лекцій [Електронний ресурс], 2018. 225 с.
4. Л.С. Глоба. Розробка інформаційних ресурсів та систем (Том 1: «Розподілені системи», «Розподілені системи. Поняття розподіленого середовища», «Зв'язок», «Процеси», «Іменування», «Синхронізація») : Підручник. К: “Політехніка”, 2013. 378 с.
5. Л.С. Глоба. Розробка інформаційних ресурсів та систем (Том 2: «Несуперечливість і реплікація», «Відмовостійкість», «Захист інформації», «Розподілені системи об'єктів», «Розподілені файлові системи», «Розподілені системи документів», «Розподілені системи узгодження», «Пошукові системи»): Підручник. К: “Політехніка”, 2013. 378 с.
6. О.В. Олецький. Перші кроки в JAVA: Навчальний посібник. Київ. 2017. 144 с.
7. К.К. Кадомський, П.К. Ніколюк. JAVA.ТЕОРІЯ І ПРАКТИКА : Навчальний посібник. Вінниця: Донну, 2019. 197 с.
8. В.Д. Тарарака. Архітектура комп'ютерних систем : Навчальний посібник. Житомир : ЖДТУ, 2018. 383 с.

4. ПИТАННЯ ДО ЗАХОДІВ ПОТОЧНОГО ТА ПІДСУМКОВОГО КОНТРОЛЮ

4.1. Тестові завдання до модульної контрольної роботи модуля ЗМ-Л1

1. У чому відмінності понять «процес» і «потік»? [1, с.8-40]
2. Опишіть призначення методу `int read()`. Чому метод повертає тип `int` при читанні байтових і символьних потоків? [1, с.44-55]
3. Проведіть порівняльний аналіз концепцій байтових потоків `InputStream / OutputStream` і байтових масивів. [1, с.44-55]
4. Особливості менеджера компоновання `FlowLayout`. [6, с.169-171]
5. Опишіть, як реалізується багатопотокові програми в Java за допомогою класу `Thread` і інтерфейсу `Runnable`. Наведіть дві структурні схеми побудови поточкових додатків на основі реалізації інтерфейсу `Runnable` і успадкування класу `Thread`. [1, с.8-14]
6. Опишіть призначення методів `int read (byte [] b)` і `int read (char [] c)`. Чому методи повертають тип `int`? [1, с.48-55]
7. Які конструктори і методи містить клас? Чи можна здійснювати читання і запис з/в файл методами даного класу? [1, с.44-48]
8. Опишіть методи класу `Thread`. Наведіть приклади їх використання. [1, с.8-14]
9. Опишіть призначення методу `flush()`. В яких випадках його слід застосовувати? [1, с.48-60]
10. Які класи (інтерфейси) пакета `java.io` розроблені відповідно до патерну `Decorator`. Їх призначення? [1, с.44-60]
11. Особливості менеджера компоновання `BorderLayout`. [6, с.167-169]
12. Опишіть методи інтерфейсу `Runnable`. Які особливості роботи з методом `run()`? [1, с.8-14]
13. Опишіть призначення методу `void write (int v)`. Поясніть, чому метод приймає тип `int` при виконанні запису байтових і символьних потоків? [1, с.44-55]
14. Які класи (інтерфейси) пакета `java.io` розроблені відповідно до патерном `Adapter`. Їх призначення? [1, с.44-60]
15. Особливості менеджера компоновання `CardLayout`. [6, с.174-176]
16. Наведіть програмний код, який обчислює суму певного виду... в двох потоках. [1, с.8-40]
17. Наведіть програмний код застосунку, у вікні якого розмістити три елементи управління `TextField` і список `List` з пунктами «СУМА» і «РІЗНИЦЯ». Додаток повинен обчислювати суму або різницю чисел відповідно до обраного пункту списку. [1, с.8-40]
18. Наведіть програмний код методу, який в якості параметру отримує шлях до каталогу, і виводить на екран список його вмісту. [1, с.44-48]

4.1. Тестові завдання до модульної контрольної роботи модуля ЗМ-Л2

1. Назвіть відмінності між Set і List? Між List і Map? Чому інтерфейс Map не спадкує інтерфейс Collection? [1, с.60-72]
2. Напишіть програму, в якій створіть ArrayList. Запишіть туди 10 випадкових цілих чисел. Виконайте з елементами списку певні дії (за варіантом). Виведіть результат на екран. [1, с.60-66]
3. Клас InetAddress. Призначення, схожість і відмінності методів `getByName()` і `getLocalHost()`. Приклади використання. [1, с.74-78]
4. Опишіть процедуру встановлення з'єднання між клієнтом і сервером по протоколу TCP засобами Java. [1, с.80-84]
5. Наведіть програмний код методу, який виводить на екран певні IP адреси (за варіантом) [1, с.74-78]
6. Назвіть відмінності між ArrayList і LinkedList? [1, с.60-66]
7. Наведіть програмний код, який створює LinkedList. Запишіть туди 10 випадкових цілих чисел. Виконайте з елементами списку певні дії (за варіантом). Виведіть результат на екран. [1, с.60-66]
8. Призначення та основні можливості класу URL. [1, с.74-78]
9. Опишіть процедуру передачі дейтаграмних пакетів засобами Java. [1, с.84-86]
10. Наведіть програмний код метода, що виводить на екран IP адреса і доменне ім'я локального комп'ютера. [1, с.74-80]
11. Опишіть особливості застосування інтерфейсу Set. [1, с.60-66]
12. Клас InetAddress. Перерахуйте існуючі для класу фабричні методи. [1, с.74-78]
13. Класи Socket і ServerSocket. Призначення і особливості застосування. [1, с.80-84]
14. Поясніть призначення класів колекцій в Java. [1, с.60-66]
15. Абстрактний клас URLConnection. Призначення і основні методи.[1, с.74-78]
16. Класи DatagramSocket і DatagramPacket. Призначення і особливості застосування. [1, с.84-86]
17. Наведіть програмний код клієнт – серверного додатку, що працює за протоколом TCP. [1, с.80-84]
18. Наведіть програмний код клієнт – серверного додатку, що працює за протоколом UDP. [1, с.84-86]

4.3. Тестові завдання до залікової контрольної роботи

1. Який абстрактний клас є суперкласом для всіх класів, що використовуються для читання байтів? [1, с.48-55]
2. Які з наступних операцій можна виконати над файлом за допомогою методів об'єкта класу File? [1, с.44-48]

3. Для запису рядків в текстовий файл у якому порядку и як слід використовувати класи? [1, с.44-55]
4. Які з фрагментів коду створять об'єкт класу ArrayList і додадуть елемент? [1, с.60-66]
5. Який інтерфейс реалізує клас Hashtable? [1, с.60-66]
6. Які з наведених нижче назв є іменами класів пакета java.util? [1, с.60-66]
7. Яким способом буде підключений об'єкт socket, якщо він оголошений наступним чином? [1, с.78-82]
8. Як помістити повідомлення в сокет? [1, с.78-82]
9. Виберіть зі списку класи - декоратори. [1, с.48-60]
10. Який з перерахованих інтерфейсів не є колекцією, тобто не є спадкоємцем Collection? [1, с.60-62]
11. Який клас є реалізацією інтерфейсу List у вигляді двонаправленого списку, де кожен елемент структури містить покажчики на попередній і наступний елементи? [1, с.60-66]
12. Який інтерфейс найбільш придатний для створення класу для зберігання даних у вигляді пари ключ/значення? [1, с.60-66]
13. Методи якого класу дозволяють організувати читання байтів з файлу? [1, с.48-55]
14. Дані якого типу повертає метод read() абстрактного класу InputStream? [1, с.48-55]
15. Як правильно додати значення в створену в такий спосіб хеш таблицю? [1, с.66-72]
16. Що повертає метод hashCode ()?[1, с.66-72]
17. Який з перелічених методів класу InetAddress Повертає масив усіх адрес, що закріплені за хостом? [1, с.74-78]
18. З наведенням фрагментів кодів тільки один правильний. Визначте який саме. [1, с.44-60]
19. Сокетом називається набір наступний ідентифікуючих параметрів...[1, с.80-81]
20. Клас java.net.Socket реалізує функції ... [1, с.80-84]
21. Який абстрактний клас є суперкласом для всіх класів, що використовуються для запису байтів? [1, с.48-55]
22. Які з наступних операцій не можна виконати над файлу за допомогою методів об'єкта класу File? [1, с.44-48]
23. Для читання рядків з текстового файлу у якому порядку и як слід використовувати класи? [1, с.44-48]
24. Який інтерфейс найбільш придатний для створення класу, що містить несортовані унікальні об'єкти? [1, с.60-66]
25. Які з наведених нижче назв є іменами інтерфейсів пакета java.util? [1, с.60-66]
26. Яким способом буде підключений об'єкт socket, якщо він оголошений наступним чином...?[1, с.80-84]
27. Як отримати вміст сторінки, використовуючи його URL? [1, с.74-78]

28. Які виняткові ситуації можливі при відкритті сокетного з'єднання? [1, с.78-82]
29. Як закрити сокети з'єднання після передачі інформації? [1, с.80-82]
30. Виберіть зі списку класи - адаптери. [1, с.44-60]
31. Який клас є реалізацією інтерфейсу List у вигляді динамічного масиву? [1, с.60-66]
32. Методи якого класу дозволяють організувати читання з байтового масиву? [1, с.48-55]
33. Дані якого типу повертає метод read() класу Reader? [1, с.48-55]
34. Яке значення повертає метод read() класу InputStream в разі, коли немає більше даних для читання? [1, с.48-55]
35. Як правильно додати значення в створену хеш картку? [1, с.66-72]
36. Що повертає метод hashCode()? [1, с.66-72]
37. Який з перелічених методів класу InetAddress повертає об'єкт класу InetAddress для комп'ютера DNS - і'мя якого передається в метод? [1, с.74-78]
38. З наведених фрагментів кодів тільки один правильний. Визначте який саме. [1, с.60-66]
39. Клас java.net.ServerSocket реалізує функції ... [1, с.80-84]
40. Як вилучити повідомлення з сокету? [1, с.80-84]

5. ЛІТЕРАТУРА ДЛЯ ВИВЧЕННЯ ДИСЦИПЛІНИ

5. ЛІТЕРАТУРА ДЛЯ ВИВЧЕННЯ ДИСЦИПЛІНИ

Основна література

1. Кузніченко С.Д., Терещенко Т.М. Крос-платформне програмування. Конспект лекцій. – Одеса: ОДЕКУ, 2016. – 104 с.

Додаткова література

2. Степанов Е.О., Ярцев Б.М. Навчально-методичний посібник з дисципліни «Архітектури та технології розробки розподіленого програмного забезпечення», 2016. – 103 с. Режим доступу: <https://books.ifmo.ru/file/pdf/425.pdf>

3. Блінов І. Н. Java. Промислове програмування: практ. посіб. / І. М. Блінов, В. С. Романчик. Видавництво: УніверсалПрес, 2007. – 768 с.

4. Кулямін В.В. Технології програмування. Компонентний похід Інтернет-університет інформаційних технологій – БІНОМ. Лабораторія знань, 2007. – 316 с. – Режим доступу: <http://panda.ispras.ru/~kuliamin/lectures-sdt/sdt-book-2006.pdf>

5. Степанов Е.О. Крос-платформні та багатоланкові технології. / Інтернет-університет інформаційних технологій – ІНТУІТ, 2010 = режим доступу: <https://www.intuit.ru/studies/courses/571/427/info>

6. Портянкін І., Swing: популяція поширених інтерфейсів. - Видавництво «Лорі», 2011. – 591 с. Режим доступу: https://ipsoftware.ru/pdf/Swing_2nd_ed_Ivan_P.pdf

7. Офіційний сайт Oracle. The Java Tutorials [електронний ресурс]. Режим доступу: <http://docs.oracle.com/javase/tutorial/index.html>

8. Horstmann Cay S., Cornell Gary. Core Java 2: Volume I–Fundamentals. Publisher: Prentice Hall PTR 2000. – 832 p.

9. Horstmann Cay S., Cornell Gary. Java 2. Library of the professional. Java 2. Volume 2. Subtleties of programming, 7th ed. "Williams", 2007. - 1168 p.

10. Васильєв А. Н. Java. Об'єктно-орієнтоване програмування: Навчальний посібник. Видавництво: Рига-Прес, 2011. – 400 с.

11. Joshua Bloch. Java. Effective Java. Publisher: Addison-Wesley Professional, 2017. – 416 p.

12. Mark Grand. Patterns in Java-Vol.1. Publisher : Wiley, 1998 – 480 p.

13. Gamma, E., Helm, R., Johnson, R. and Vlissides, J. Design Patterns: Elements of Reusable Object-Oriented Software. Addison-Wesley, Boston. 1994 – 417 p. Режим доступу: <http://www.javier8a.com/itc/bd1/articulo.pdf>

14. Patrick Naughton, Herbert Schildt. Java 2: The Complete Reference. Publisher : McGraw Hill, 1999. 1108 p.

15. Steve McConnell. Code Complete 1st Edition. Publisher : Microsoft Press; 1993. 857 p.
16. Harvey M. Deitel, Paul J. Deitel, S. E. Santry. Advanced Java 2 Platform: How to Program. Publisher : Prentice Hall; 2001. 1811 p.
17. David Geary. Core JavaServer Faces. Publisher:Prentice Hall,2010. 656 p.
18. Bruce Eckel. Thinking in Java. Publisher : Pearson; 2006. 1150 p. Режим доступу: https://sd.blackball.lv/library/thinking_in_java_4th_edition.pdf
19. Електронна бібліотека ОДЕКУ www.library-odeku.16mb.com

Методичні вказівки

20. *Кузнiченко С.Д.* Методичні вказівки до виконання лабораторних робіт з дисципліни “Крос-платформне програмування” для студентів 4 курсу денної форми навчання. Напря́м підготовки – комп’ютерні науки. – Одеса: ОДЕКУ, 2015. – 153 с.
21. *Кузнiченко С.Д.* Методичні вказівки до виконання самостійної роботи студентів з дисципліни “Крос-платформне програмування” для студентів 4 курсу денної форми навчання. Напря́м підготовки – комп’ютерні науки. – Одеса: ОДЕКУ, 2016. – 96 с.