

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Одеський державний екологічний університет

ЗАТВЕРДЖЕНО

на засіданні групи забезпечення спеціальності
від « 20 » 05 2021 року
протокол № 13
Голова групи д.т.н., проф. Мещеряков В.І.

УЗГОДЖЕНО

Декан факультету комп'ютерних наук,
управління та адміністрування
к.геогр.н., доцент Кузніченко С.Д.

СИЛЛАБУС

навчальної дисципліни

Об'єктно-орієнтоване програмування

(назва навчальної дисципліни)

122 – «Комп'ютерні науки»

(шифр та назва спеціальності)

«Комп'ютерні науки»

(назва освітньої програми)

РВО «Бакалавр»

(рівень вищої освіти)

денна

(форма навчання)

2 рік н.

(рік навчання)

4 семестр

(семестр навчання)

4 кр./120 год.

(кількість кредитів ЄКТС/годин)

іспит

(форма контролю)

Інформаційних технологій

(кафедра)

Одеса, 2021 р.

Автори: Кузнiченко С.Д., к.геогр.н., доцент,
Клепатська В.В., асистент

Поточна редакція розглянута на засiданнi кафедри iнформацiйних технологiй вiд
18.05.2021 року, протокол № 9.

Викладачi: лекцiйнi заняття – Кузнiченко Свiтлана Дмитрiвна, к.геогр.н., доцент
лабораторнi заняття – Клепатська Вiкторiя Вiкторiвна, асистент.

Перелiк попереднiх редакцiй

Прiзвища та iнiцiали авторiв	Дата, № протоколу	Дата набуття чинностi

1. ОПИС НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Мета	Формування у здобувачів вищої освіти практичних навичок в використанні об'єктно-орієнтованої парадигми проектування та розробки програмної логіки із застосуванням сучасних технологій та бібліотек зі створення програмних застосувань на базі мови програмування Java.
Компетентність	<p>СК3. Здатність до логічного мислення, побудови логічних висновків, використання формальних мов і моделей алгоритмічних обчислень, проектування, розроблення й аналізу алгоритмів, оцінювання їх ефективності та складності, розв'язності та нерозв'язності алгоритмічних проблем для адекватного моделювання предметних областей і створення програмних та інформаційних систем</p> <p>СК8. Здатність проектувати та розробляти програмне забезпечення із застосуванням різних парадигм програмування: узагальненого, об'єктно-орієнтованого, функціонального, логічного, з відповідними моделями, методами й алгоритмами обчислень, структурами даних і механізмами управління</p>
Результат навчання	<p>ПР5. Проектувати, розробляти та аналізувати алгоритми розв'язання обчислювальних та логічних задач, оцінювати ефективність та складність алгоритмів на основі застосування формальних моделей алгоритмів та обчислюваних функцій</p> <p>ПР10. Використовувати інструментальні засоби розробки клієнт-серверних застосувань, проектувати концептуальні, логічні та фізичні моделі баз даних, розробляти та оптимізувати запити до них, створювати розподілені бази даних, сховища та вітрини даних, бази знань, у тому числі на хмарних сервісах, із застосуванням мов веб-програмування</p>
Базові знання	<p>1. Базові принципи об'єктно-орієнтованого програмування (абстракція, інкапсуляція, успадкування та поліморфізм);</p> <p>2. Базові знання мови програмування Java;</p> <p>3. Знання інтегрованих середовищ розробки програмних застосувань.</p>
Базові вміння	<p>1. Вміння практичного застосування принципів ООП;</p> <p>2. Вміння використання мови програмування Java</p>

	для розробки об'єктно-орієнтованих програмних застосувань.
Базові навички	Базові навички з розробки програмної логіки із застосуванням об'єктно-орієнтованої парадигми проектування та сучасних технологій і бібліотек зі створення програмних застосувань на базі мови програмування Java
Пов'язані силлабуси	–
Попередня дисципліна	Алгоритмізація та програмування
Наступна дисципліна	–
Кількість годин	лекції: 30 год.; лабораторні заняття: 30 год; самостійна робота студентів: 60 год.

2. ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

2.1 Лекційні модулі

Код	Назва модуля та тем	Кількість годин	
		аудиторні	СРС
ЗМ-Л1	<i>Принципи об'єктно-орієнтованого програмування</i>		
	• Основи ООП	1	3+5
	• Абстрагування даних та інкапсуляція	1	
	• Конструктори і класів	2	
	• Перевантаження конструкторів та методів	2	
	• Статичні, константні члени класів	2	
	• Просте успадкування класів	4	
	• Композиція об'єктів супроти успадкування	2	
ЗМ-Л2	<i>Інтерфейси та вкладені класи. Оброблення виняткових ситуацій</i>	2	3+5
	• Реалізація поліморфізму	4	
	• Інтерфейси та множинне успадкування	2	
	• Вкладені та внутрішні класи	2	
	• Оброблення виняткових ситуацій	6	
	• Параметризовані класи та методи		
Підготовка до іспиту			20
Разом:		30	36

Консультації:

Кузніченко Світлана Дмитрівна, **понеділок 12⁴⁵ – 14²⁰ год.** 320 ауд. НЛК№1

2.2 Практичні модулі

Код	Назва модуля та тем	Кількість годин	
		аудиторні	СРС
ЗМ-П1	1) Класи та об'єкти. Способи ініціювання полів.	4	3
	2) Застосування конструкторів ініціювання та копіювання об'єктів, перевантаження методів.	4	3
	3) Спадкування членів класу. Перевизначення методів.	4	3
	4) Абстрактні класи та їх реалізація.	4	3
ЗМ-П2	1) Опис та реалізація інтерфейсів.	4	4
	2) Обробка винятків, що виникають під час виконання програми.	4	4
	3) Внутрішні класи та їх реалізація	6	4
Разом:		30	24

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 12⁴⁵ – 14²⁰ год.** 330 ауд. НЛК№1

2.3 Самостійна робота студента та контрольні заходи

Код	Завдання на СРС та контрольні заходи	Кількість годин	Строк проведення
ЗМ-Л1	• Підготовка до лекційних занять	3	1-7 тижні
	• Підготовка до модульної контрольної роботи № 1	5	1-7 тижні
	• Модульна контрольна робота № 1 (обов'язкова)		7 тиждень
ЗМ-Л2	• Підготовка до лекційних занять	3	8-15 тижні
	• Підготовка до модульної контрольної роботи № 2	5	8-15 тижні
	• Модульна контрольна робота № 2 (обов'язкова)		15 тиждень
ЗМ-П1	• підготовка до усного опитування напередодні відповідної лабораторної роботи (обов'язкове)	4	1-8 тижні
	• підготовка до захисту звіту з лабораторних робіт (обов'язковий)	8	1-8 тижні

ЗМ-П2	• підготовка до усного опитування напередодні відповідної лабораторної роботи (обов'язкове)	3	9-15 тижні
	• підготовка до захисту звіту з лабораторних робіт (обов'язковий)	9	9-15 тижні
	Підготовка до іспиту	20	
Разом:		60	

1. Методика проведення та оцінювання контрольного заходу для ЗМ-Л1

Контроль проводиться після вивчення лекційного матеріалу модуля ЗМ-Л1 в формі письмової модульної контрольної роботи МКР-1 в якій студенти відповідають на 10 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 25 балів; розподіл балів за правильні відповіді має наступний вигляд:

Номер запитання	Кількість балів за правильну відповідь
1	2
2	2
3	2
4	2
5	2
6	2
7	2
8	4
9	4
10	5
Разом:	25

Критерії оцінювання результатів контрольного заходу: відмінно (22-25 балів); добре (19-21 балів); задовільно (15-18); незадовільно (менше 15 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 10 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 25 балів. Розподіл балів за правильні відповіді має наступний вигляд:

Номер запитання	Кількість балів за правильну відповідь
1	2
2	2
3	2
4	2
5	2
6	2
7	2
8	4
9	4
10	5
Разом:	25

Критерії оцінювання результатів контрольного заходу: відмінно (22-25 балів); добре (19-21 балів); задовільно (15-18); незадовільно (менше 15 балів).

3. 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, 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.

4. Методика проведення та оцінювання контрольного заходу для ЗМ-П1

За практичний модуль ЗМ-П1 встановлена максимальна оцінка 20 балів. ЗМ-П1 складається з чотирьох лабораторних робіт, за якими встановлена максимальна оцінка 5 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 4, максимальна кількість балів – 2),
- *захисту результатів* лабораторної роботи наведених у звіті до лабораторної роботи (кількість запитань залежить від ходу виконання студентом роботи і якості звіту, максимальна кількість балів – 3).

Для кожної лабораторної роботи, якщо студент за *усне опитування* одержав менше 1 балу 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 практичний модуль ЗМ-П1 буде сума балів за всі лабораторні роботи. Критерії оцінювання результатів контрольного заходу для ЗМ-П1: 18 балів і більше – відмінно, 15...17,9 – добре, 12...14,9 балів – задовільно, менше 12 балів – незадовільно.

5. Методика проведення та оцінювання контрольного заходу для ЗМ-П2

За практичний модуль ЗМ-П2 встановлена максимальна оцінка 30 балів. ЗМ-П1 складається з трьох лабораторних робіт, за якими встановлена максимальна оцінка 10 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 4, максимальна кількість балів – 2),
- *захисту результатів* лабораторної роботи наведених у звіті до лабораторної роботи (кількість запитань залежить від ходу виконання студентом роботи і якості звіту, максимальна кількість балів – 8).

Для кожної лабораторної роботи, якщо студент за *усне опитування* одержав менше 1 балу 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 практичний модуль ЗМ-П2 буде сума балів за всі лабораторні роботи. Критерії оцінювання результатів контрольного заходу для ЗМ-П2: 17 балів і більше – відмінно, 22,5...16,9 – добре, 18...22,94 балів – задовільно, менше 18 балів – незадовільно.

6. Методика оцінювання за всіма змістовними модулями

Підсумковою оцінкою за всіма змістовними модулями (ОЗ) буде сума балів за лекційні модулі та практичні модулі.

7. Методика проведення та оцінювання підсумкового контрольного заходу

Підсумковий контрольний захід проводиться у формі іспиту. Студент вважається допущеним до підсумкового контролю з навчальної дисципліни, якщо він виконав всі види робіт поточного контролю, передбачені робочою навчальною програмою дисципліни і набрав за накопичувальною системою суму балів не менше 25 балів з практичної частини курсу.

Результати роботи оформлюються на окремому аркуші. Час, що виділяється на виконання підсумкового контрольного заходу визначається при видачі завдання і не перевищує 1 академічної години.

Оцінювання письмових відповідей студентів проводиться у відповідності з Положенням «Про критерії оцінки знань студентів в ОДЕКУ». Загальна екзаменаційна оцінка (бал успішності) еквівалента відсотку правильних відповідей із загального обсягу питань екзаменаційного білету, який складає 20 питань. Тобто кожне питання оцінюється в 5 балів. Максимальна сума балів становить 100.

Таблиця для визначення кількісних показників успішності за підсумками іспиту

За системою університету (у відсотках)	Визначення
90 – 100	відмінне виконання лише з незначною кількістю помилок
82 – 89	вище середнього рівня з кількома помилками
74 – 81	в загальному правильна робота з певною кількістю грубих помилок
64 – 73	непогано, але зі значною кількістю помилок
60 – 63	виконання задовольняє мінімальні критерії
35 – 59	з можливістю перескласти
1 – 34	з обов'язковим повторним курсом навчання

8. Методика підсумкового оцінювання за дисципліну

Накопичена підсумкова оцінка засвоєння студентом навчальної дисципліни, що закінчується іспитом, розраховується як:

$$ПО = 0,5ОПК + 0,5ОЗ. \quad (1)$$

де: ОПК – кількісна оцінка (у відсотках від максимально можливої) заходу підсумкового контролю;

ОЗ – кількісна оцінка (у відсотках від максимально можливої) заходів за всіма змістовними модулями.

Шкала відповідності оцінок в системах університету, національній системі та системі ЄКТАС наведена в таблиці.

За шкалою ECTS	За національною системою	Бал успішності
	для іспиту	
A	5 (відмінно)	90–100
B	4 (добре)	82–89,9
C	4 (добре)	74–81,9
D	3 (задовільно)	64–73,9
E	3 (задовільно)	60–63,9
FX	2 (незадовільно)	35–59,9
F	2 (незадовільно)	1–34,9

3. РЕКОМЕНДАЦІЇ ДО САМОСТІЙНОЇ РОБОТИ СТУДЕНТІВ

Рекомендується наступний порядок вивчення дисципліни «Об'єктно-орієнтоване 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.

Модуль ЗМ-Л1 «Принципи об'єктно-орієнтованого програмування »

3.1.1. Повчання

Розділи модуля ЗМ-Л1 формують у студентів уявлення про загальні принципи об'єктно-орієнтованого програмування: про абстрагування даних та інкапсуляцію, конструктори і методи класів та їх перевантаження, модифікатори доступу членів класів, успадкування класів, агрегацію і композицію.

При вивченні цих розділів необхідно звернути увагу на викладання основних принципів об'єктно-орієнтованого програмування: абстракції, інкапсуляції, успадкування та їх реалізацію в Java.

Програма ЗМ-Л1. Принципи об'єктно-орієнтованого програмування

Основи ООП [1, с.7-18]

Абстрагування даних та інкапсуляція [1, с.64-81]

Конструктори і класів [1, с.36-44; с.47-51]

Перевантаження конструкторів та методів [1, с.58-75]

Статичні, константні члени класів [1, с.45-47]

Просте успадкування класів [1, с.93-109]

Композиція об'єктів супроти успадкування [1, с.110-123]

3.1.2. Питання для самоперевірки

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

1. Загальні положення об'єктно-орієнтованої парадигми програмування
2. Основні поняття ООП
3. Збереження даних у пам'яті
4. Поняття і властивості об'єкта, оголошення простого класу на мові JAVA
5. Модифікатори оголошення класу
6. Поля класу їх модифікатори та конструкція ініціювання
7. Статичні поля та поле final
8. Управління доступом до членів класу, модифікатори доступу
9. Створення об'єктів
10. Блоки ініціювання та ініціалізація статичних полів
11. Конструктори їх відміна від методів класу
12. Методи класу, їх оголошення та модифікатори
13. Виклик методів, передача та повернення параметрів
14. Застосування методів для керування доступом
15. Вираз this і його застосування
16. Перевантаження методів, метод main та методи native
17. Спадкування, супер та субкласи
18. Конструктори субкласів
19. Порядок ініціювання полів субкласів
20. Перевизначення методів класу
21. Приховування полів та доступ до успадкованих членів
22. Службове слово super і його застосування
23. Сумісність, перетворення та перевірка типів
24. Особливості застосування модифікаторів protected та final
25. Модифікатор методів abstract і абстрактні класи
26. Клас Object і його методи
27. Клонування об'єктів
28. Типи даних, що перелічуються (enum)

3.2. Модуль ЗМ-Л2 «Інтерфейси та вкладені класи. Оброблення виняткових ситуацій»

3.2.1. Повчання

Розділи модуля ЗМ-Л2 формують у студентів уявлення про успадкування класів та композицію об'єктів, реалізацію поліморфізму, вкладені і внутрішні класи, параметризовані класи та методи, оброблення виняткових ситуацій.

При вивченні цих розділів необхідно звернути увагу на особливості реалізації поліморфізму та оброблення виняткових ситуацій на мові програмування JAVA.

Програма ЗМ-Л2. Інтерфейси та вкладені класи. Оброблення виняткових ситуацій

Реалізація поліморфізму [1, с.124-157]

Інтерфейси та множинне успадкування [1, с.158-184]

Вкладені та внутрішні класи [1, с.185-206]

Оброблення виняткових ситуацій [1, с.207-230]

Параметризовані класи та методи [1, с.231-251]

3.2.2. Питання для самоперевірки

Запитання, що входять до тестів до модуля ЗМ-Л2 і являють собою необхідний мінімум знань, який потрібний для засвоєння дисципліни «Об'єктно-орієнтоване програмування», наведені нижче:

1. Наслідування або композиція
2. Принципи проектування класу, що підлягає спадкуванню
3. Поняття інтерфейсу, стандартні інтерфейси мови JAVA
4. Оголошення інтерфейсу, константи та методи у ньому
5. Розширення інтерфейсів, спадкування та приховування констант
6. Спадкування, пере визначення і перевантаження методів у інтерфейсах
7. Робота з інтерфейсами та їх реалізація
8. Пусті інтерфейси та правила застосування інтерфейсів
9. Винятки та їх види
10. Створення типів винятків
11. Інструкція throw, передача управління
12. Висловлювання throws і пере визначення методів
13. Блок try
14. Висловлювання finally
15. Правила застосування винятків
16. Статичні вкладені класи і інтерфейси
17. Внутрішні класи
18. Доступ до полів внутрішніх та зовнішніх класів
19. Спадкування внутрішніх класів
20. Спадкування і приховування у контексті зовнішнього класу
21. Локальні внутрішні класи
22. Анонімні локальні класи
23. Вкладеність у інтерфейсах та зміни в них

3.3. Модуль ЗМ-П1 „Практичний модуль”

При вивченні практичного модуля студенти набувають уміння описувати класи, що містять статичні й нестатичні компоненти, задавати значення полів за допомогою блоків ініціалізації, створювати об'єкти і масиви об'єктів, розробляти різні типи конструкторів, будувати класи, що використовують перевантажені методи [17,18].

При вивченні цього модуля необхідно звернути увагу на практичне застосування одержаних теоретичних знань про ієрархію спадкування, використання абстрактних класів та реалізації таких класів і методів 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.

3.4. Модуль ЗМ-П2 „Практичний модуль”

При вивченні практичного модуля студенти набувають уміння описувати та застосовувати інтерфейси, створювати класи, що реалізують інтерфейс і працювати з об'єктами типу інтерфейс [19].

При вивченні цього модуля необхідно звернути увагу на практичне застосування одержаних теоретичних знань про механізми обробки виняткових ситуацій в Java та принципи побудови класів, що містять у своєму складі внутрішні класи.

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.

4. ПИТАННЯ ДО ЗАХОДІВ ПОТОЧНОГО ТА ПІДСУМКОВОГО КОНТРОЛЮ

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

1. Який модифікатор класу означає неможливість успадкування від цього класу? [1, с.23-25; с.93-110]
2. Який модифікатор поля означає його приналежність контексту класу, а не об'єкту? [1, с.45-47]
3. Скільки класів може розширювати (бути безпосереднім нащадком) клас в Java? [1, с.93-110]
4. За допомогою якого ключового слова з методу дочірнього класу можна викликати перевизначений метод батьківського класу? [1, с.93-110]
5. Як називається принцип ООП, постулює можливість заміни об'єктів з схожим інтерфейсом? [1, с.8-25]
6. Виберіть зі списку основні принципи ООП [1, с.8-25]
7. Виберіть методи класу Object, які можна перевизначити при спадкуванні [1, с.140-150]
8. Фрагмент коду створює один об'єкт Point і один об'єкт Rectangle. Скільки посилань на ці об'єкти існує після того, як код буде виконаний? Чи буде який-небудь з об'єктів кандидатом для «збірки сміття»? ... [1, с.36-45]
9. Приховування реалізації класу і відділення його внутрішнього уявлення від зовнішнього називається... [1, с.8-25]
10. Відношення між класами, при якому один клас використовує структуру і поведінку іншого або інших класів називається... [1, с.8-25]

11. Використовуючи поняття сигнатури методу, визначте, які методи несумісні один з одним в класі [1, с.58-72]
12. Визначте, що буде виведено на екран при компіляції і запуску наступного коду:

```
public class Example {  
    static int i;  
    public static void main (String [] args) {  
        System.out.print (i);  
    }  
}
```

[1, с.45-47]
13. Визначте, що буде виведено на екран при компіляції і запуску наступного коду:

```
class My {  
    int x, y;  
    My (int x, int y) {this.x = x; this.y = y;}  
    My (int x) {this (x, -2);}  
    My () {this (1,2);}  
    public static void main (String [] args) {  
        My a = new My (5);  
        System.out.println ((a.x + a.y)); }  
}
```

[1, с.47-75]
14. Маючи клас з назвою Point, напишіть деякий код, який створює екземпляр цього класу, ініціалізує його дві змінні, і потім виводить на екран значення кожної зі змінних. [1, с.47-75]
15. Визначте клас Point, що задає точку на площині. Клас повинен містити кілька конструкторів, реалізувати такі методи для обчислення відстані від точки до початку координат та відстані між двома точками. [1, с.47-75]
16. Визначте клас Matrix розмірності (mxn). Клас повинен містити кілька конструкторів. Оголосіть масив об'єктів. Передайте об'єкти в метод, що знаходить суму елементів. [1, с.47-75]
17. Дано клас...:
Оберіть правильні варіанти успадкування від цього класу. [1, с.93-110]
18. Реалізуйте на Java наведену нижче ієрархію класів [1, с.93-110]
19. Реалізуйте на Java наведену нижче ієрархію класів, зробив деякі класи абстрактними [1, с.154-158]
20. Створіть клас Person, що враховує принцип інкапсуляції і містить наступні поля: ім'я, адреса та вік людини. [1, с.22-24]

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

1. За допомогою якого ключового слова можна примусово викинути виняток? [1, с.207-230]
2. Які блоки в конструкції try/catch/finally можуть бути відсутні? [1, с.207-230]
3. Яким модифікатором неявно забезпечуються всі елементи інтерфейсу? [1, с.158-178]

4. Реалізуйте на Java наведену нижче ієрархію класів з використанням інтерфейсів [1, с.158-178]
5. Здатність об'єкта відповідати під час виконання двох або більш можливим типам називається... [1, с.124-150]
6. Яке службове слово треба вказати, якщо клас реалізує інтерфейс? [1, с.158-178]
7. Дайте визначення поняттю "виняток"? [1, с.207-211]
8. Яка ієрархія винятків? [1, с.207-211]
9. Можна / чи потрібно обробляти помилки jvm? [1, с.212-224]
10. Про що говорить ключове слово throws? [1, с.212-224]
11. Чи може один блок catch відловлювати кілька винятків (з одного і різних гілок спадкувань)? [1, с.211-224]
12. Якщо оператор return міститься і в блоці catch і в finally, який з них "головніше"? [1, с.211-224]
13. Дайте визначення поняттю "інтерфейс" [1, с.158-176]
14. Чому не можна оголосити метод інтерфейсу з модифікатором final або static? [1, с.158-175]
15. Які типи класів бувають в java [1, с.187-198]
16. Які особливості створення вкладених класів: простих і статичних [1, с.203-207]
17. Які класи називаються анонімними? [1, с.198-203]
18. Яким чином з вкладеного класу отримати доступ до поля зовнішнього класу? [1, с.203-207]
19. У чому різниця вкладених і внутрішніх класів? [1, с.185-207]
20. Яким чином можна звернутися до локальної змінної методу з анонімного класу, оголошеного в тілі цього методу? Чи є які-небудь обмеження для такої змінної? [1, с.198-203]

4.3. Тестові завдання до іспиту

1. Приховування реалізації класу і відділення його внутрішнього уявлення від зовнішнього називається... [1, с.18-25]
2. Здатність об'єкта відповідати під час виконання двох або більш можливим типам називається... [1, с.18-25]
3. У чому різниця вкладених і внутрішніх класів? [1, с.185-206]
4. Як пов'язаний будь-який користувацький клас з класом Object? [1, с.140-154]
5. В чому відмінність оброблюваних від необроблюваних (checked/unchecked) винятків? [1, с.207-214]
6. Які існують способи обробки винятків? [1, с.214-222]
7. Як називається принцип ООП, постулює можливість заміни об'єктів з схожим інтерфейсом? [1, с.18-25]
8. Виберіть зі списку основні принципи ООП [1, с.18-25]
9. Відношення між класами, при якому один клас використовує структуру і поведінку іншого або інших класів називається... [1, с.18-25]

10. Дайте визначення поняттю "клас"[1, с.36-43]
11. Як правильно організувати доступ до полів класу? [1, с.47-51]
12. Дайте визначення поняттю "конструктор"[1, с.51-58]
13. Чим відрізняються конструктори за замовчуванням, копіювання і конструктор з параметрами? [1, с.51-84]
14. Дайте визначення узагальненим класам та методам [1, с.231-248]
15. Яка особливість класу з єдиним закритим (private) конструктором? [1, с.51-84]
16. Про що говорять, де і як можна використовувати ключове слова "super"? [1, с.74-75]
17. Що таке сигнатура методу? [1, с.58-84]
18. Які методи називаються перевантаженими? [1, с.58-84]
19. Чи можуть нестатичні методи перевантажити статичні? [1, с.45-84]
20. Що таке перевизначення методів? [1, с.58-75]
21. Чи може метод приймати різну кількість параметрів (аргументи змінної довжини)? [1, с.75-84]
22. Чи можна знизити рівень доступу/тип значення при перевизначенні методу? [1, с.24-25]
23. Як отримати доступ до перевизначення методів батьківського класу? [1, с.61-110]
24. Скільки класів може розширювати (бути безпосереднім нащадком) клас в Java? [1, с.93-110]
25. Чим відрізняється перевизначення від перевантаження? [1, с.61-110]
26. Де можна формувати статичні / нестатичні поля? [1, с.45-47]
27. Навіщо потрібен оператор instanceof? [1, с.139-142]
28. Навіщо потрібні і які бувають блоки ініціалізації? [1, с.47-51]
29. Який порядок виклику конструкторів і блоків ініціалізації двох класів: нащадка і його предка? [1, с.93-110]
30. Де і для чого використовується модифікатор abstract? [1, с.154-158]
31. Чи можна оголосити метод абстрактним і статичним одночасно? [1, с.154-158]
32. До яких конструкцій Java застосуємо модифікатор static? [1, с.45-47]
33. Що буде, якщо в static блоці коду виникне виняткова ситуація? [1, с.207-227]
34. Чи можна перевантажити static метод? [1, с.58-84]
35. Про що говорить ключове слово final? [1, с.36-47]
36. Чому не можна оголосити метод інтерфейсу з модифікатором final або static?
37. Які типи класів бувають в java ? [1, с.18-25]
38. У чому різниця вкладених і внутрішніх класів? [1, с.185-198]
39. Які класи називаються анонімними? [1, с.198-203]
40. Яким чином з вкладеного класу отримати доступ до поля зовнішнього класу? [1, с.203-207]

5. ЛІТЕРАТУРА ДЛЯ ВИВЧЕННЯ ДИСЦИПЛІНИ

Основна

1. Рольщиков В.Б. Об'єктно-орієнтоване програмування. Конспект лекцій. – Одеса: ОДЕКУ, 2020. – 268 с.
2. Верлань А.Ф., Чмырь И.А., Кузниченко С.Д., Коваленко Л.Б. Императивное программирование и объектно-ориентированное моделирование: Java, UML, OCL // Учебное пособие для студентов высших учебных заведений. Одесса «Экология», 2013 г.– 432 с.
3. Хорстманн, Кей С., Корнелл, Гари. Java2. Библиотека профессионала, том1. Основы, 8-е изд.: Пер. с англ. – М.:Издательский дом «Вильямс», 2012. – 816 с.
4. Шилдт Г. Java. Полное руководство. – 9-е изд.: Пер. с англ. – М.: ООО «И.Д. Вильямс», 2015. – 1376 с.
5. Эккель Б. Философия Java. – 4-е изд.– Санкт-Петербург: «издательский дом Питер», 2009.– 637 с.

Додаткова література

6. Официальный сайт Oracle. The Java Tutorials [электронный ресурс]. Режим доступа: <http://docs.oracle.com/javase/tutorial/index.html>
7. Васильев А. Н. Java. Объектно-ориентированное программирование: Учебное пособие. – СПб.: Питер, 2011. – 400 с.
8. Хорстманн, Кей С., Корнелл, Гари. Java2. Библиотека профессионала, том2. Тонкости программирования, 7-е изд.: Пер. с англ. – М.:Издательский дом «Вильямс», 2007. – 626 с.
9. Джошуа Блох. Java.Эффективное программирование =Effective Java. – М.: Лори, 2002. – 224 с.
10. Гранд М. Шаблоны проектирования в Java / пер. с англ. С. Беликовой. – М.: Новое знание, 2004.–. 559 с.
11. Гамма Э., Хелм Р., Джонсон Р., Влиссидес Дж. Приемы объектно-ориентированного проектирования. Паттерны проектирования. - СПб.: Питер, 2001.-368с.
12. Макконнелл С. Совершенный код. Практическое руководство по разработке программного обеспечения. 2-е изд. / Пер. с англ.- М.: Русская редакция, СПб.: Питер, 2005.- 896 с.: ил.

13. Дейтел Х. М. Технологии программирования на Java 2: Книга 2. Распределенные приложения / Х. М. Дейтел, П. Дж. Дейтел. ; пер. с англ. – М. : ООО "Бином-Пресс", 2003. – 464 с. : ил.
14. Дейтел Х. М. Технологии программирования на Java 2: Книга 3. Корпоративные системы / Х. М. Дейтел, П. Дж. Дейтел, С. И. Самтри ; пер. с англ. – М. : ООО "Бином-Пресс", 2003. – 672 с. : ил.
15. Дэвид М. Гери, JavaServer Faces. Библиотека профессионала. JavaServer Faces. CORE / Дэвид М. Гери, Кей С. Хорстманн. – 3-е изд. – М.: Издательский дом "Вильямс", 2011. – 544 с.
16. Електрона бібліотека ОДЕКУ www.library-odeku.16mb.com

Методичні вказівки

17. Рольщиков В.Б., Шуптар Н.Й. Методичні вказівки до виконання лабораторних робіт з дисципліни “Об’єктно-орієнтоване програмування. Частина 1” для студентів II курсу денної форми навчання. – Одеса: ОДЕКУ, 2013. – 38 с.
18. Рольщиков В.Б., Шуптар Н.Й. Методичні вказівки до виконання лабораторних робіт з дисципліни “Об’єктно-орієнтоване програмування. Частина 2” для студентів II курсу денної форми навчання. – Одеса: ОДЕКУ, 2013. – 30 с.
19. Рольщиков В.Б., Шуптар Н.Й. Методичні вказівки до виконання лабораторних робіт з дисципліни “Об’єктно-орієнтоване програмування. Частина 3” для студентів II курсу денної форми навчання. – Одеса: ОДЕКУ, 2015. – 57 с.