

Міністерство освіти і науки України
Харківський національний університет імені В.Н. Каразіна
Кафедра штучного інтелекту та програмного забезпечення



РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Об'єктно-орієнтоване програмування

рівень вищої освіти	<u>перший (бакалаврський) рівень</u>
галузь знань	<u>12 Інформаційні технології</u>
спеціальність	<u>122 Комп'ютерні науки</u>
освітня програма	<u>Комп'ютерні науки</u>
вид дисципліни	<u>обов'язкова</u>
факультет	<u>комп'ютерних наук</u>

Програму рекомендовано до затвердження Вченою радою факультету комп'ютерних наук
«29» серпня 2018 року, протокол № 9

РОЗРОБНИК ПРОГРАМИ:

доцент кафедри штучного інтелекту та програмного забезпечення **Поклонський Євген
Васильович**

Програму схвалено на засіданні кафедри штучного інтелекту та програмного забезпечення
Протокол від «26» червня 2018 року № 11

Завідувач кафедри штучного інтелекту та програмного забезпечення



Куклін В.М.

Програму погоджено методичною комісією факультету комп'ютерних наук
Протокол від «27» червня 2018 року № 7

Голова методичної комісії факультету комп'ютерних наук



Васильєва Л.В.



ВСТУП

Програма навчальної дисципліни «Об'єктно-орієнтоване програмування» складена відповідно до освітньо-професійної програми підготовки першого (бакалаврського) рівня вищої освіти за спеціальністю: 122 «Комп'ютерні науки»

1. Опис навчальної дисципліни

1.1. Метою вивчення курсу «Об'єктно-орієнтоване програмування» є навчити студентів самостійно розробляти невеликі програмні продукти за допомогою об'єктно-орієнтованого підходу.

1.2. Основними завданнями вивчення дисципліни є:

- навчитися використовувати об'єктно-орієнтований підхід при аналізі явищ та систем;
- досягти рівня вільного володіння переважною більшістю синтаксичних конструкцій мови Java;
- набути практичних навичок створення та відлагодження програмного кода з використанням мови Java та супутніх програмних засобів (компілятор, IDE тощо);
- познайомитися на практиці з найчастіше використовуваними пакетами платформи J2SE та отримати уявлення про наявність інших пакетів;
- познайомитися на практиці із методами аналізу, проектування ПЗ та методологіями розробки програмних систем.

1.3. Кількість кредитів – 10

1.4. Загальна кількість годин – 300

1.5. Характеристика навчальної дисципліни	
Нормативна / за вибором	
денна форма навчання	денна форма навчання
Рік підготовки	
1-й,	2-й
Семестр	
2-й	3-й
Лекції	
32 год.	32 год.
Практичні, семінарські заняття	
_____ год.	_____ год.
Лабораторні заняття	
32 год.	32 год.
Самостійна робота	
116 год.	56 год.
В т.ч. індивідуальні завдання	
5 год. (контр. робота)	__45 год. (курсовая+контр. робота)

1.6. Заплановані результати навчання

знати:

- сучасні технологій та інструментальні засоби розробки програмних систем, уміння їх застосовувати на всіх етапах життєвого циклу;
- розуміння основних індустріальних мов програмування, інформаційних систем і WEB.

вміти:

- здатність до об'єктно-орієнтованого мислення, знання об'єктно-орієнтованих мов програмування та уміння застосовувати об'єктно-орієнтований підхід під час проектуванні складних програмних систем.

2. Тематичний план навчальної дисципліни

Розділ 1. Знайомство з мовою Java. Інкапсуляція.

Тема 1. Огляд курсу

Парадигми програмування. Особливості об'єктно-орієнтованого програмування. Етапи розробки проекту. Огляд курсу.

Тема 2. Основні поняття мови Java.

Історія ООП та Java. Особливості мови Java. Структура програми. Угоди про використання імен. Види Java-додатків. Засоби розробки Java-додатків.

Тема 3. Основні поняття мови Java.

Лексичні основи мови (константи, змінні, ключові слова, операції). Типи даних: примітивні і об'єктні. Перетворення типів. Оператори мови. Типи обчислювальних процесів. Умовний оператор, умовні вирази, оператор вибору варіанту. Циклічні обчислювальні процеси (оператори циклів while, for, do-while). Особливості використання операторів циклу. Оператори break, continue

Тема 4. Структурні типи даних.

Робота з одновимірними і багатовимірними масивами. Робота зі строками.

Тема 5. Об'єктна модель Java.

Концепція об'єктно-орієнтованого програмування. JVM, JDK, CLASSPATH, компіляція та завантаження класів. Класи і об'єкти (поля, методи та конструктори.). Інкапсуляція; приклади. Рівні доступу полів. Пакети. Методи і змінні класів (приховування змінних, this, статичні методи і дані). Поєднання методів. Створення і руйнування об'єктів (конструктори, збірка "сміття", finalize ()). Статичні поля.

Розділ 2. Спадкування та поліморфізм

Тема 6. Повторне використання класів.

Композиція і успадкування. Побудова похідних класів. Ієрархічні структури класів. Структура об'єкту-спадкоємця. Порядок визову конструкторів, розрішення конфліктів імен. Приклади.

Тема 7. Об'єктна модель в Java.

Абстрактні класи. Клас Object. Динамічне зв'язування (поліморфізм). Приклади поліморфного поведінку. Посилання типу суперкласу. Пряме та зворотне приведення типів. Раннє та пізнє зв'язування.

Тема 8. Об'єктна модель в Java.

Проблема множинного наслідування. Інтерфейси. Ієрархія інтерфейсів. Порівняння абстрактних класів і інтерфейсів.

Розділ 3. Обробка виключень, внутрішні класи і огляд стандартних пакетів.**Тема 9. Внутрішні класи.**

Внутрішні класи в методі і контексті. Анонімний внутрішній клас. Зв'язок із зовнішнім класом.

Тема 10. Обробка виключень.

Загальні відомості про обробку виключень. Класи виключень. Механізм обробки виключень (try, catch, throw, throws; блок finally). Кілька розділів catch. Вкладені оператори try.

Тема 11. Пакет java.lang.

Оболонки примітивних типів. Клас Runtime. Клас System. Клас Class.

Тема 12. Сервісні класи пакету java.util.

Розбір строк (StringTokenizer). Класи для роботи з датами. Робота з часом.

Тема 13. Пакети і утиліти.

Пакети java.math, java.text. Документація і утиліта java.doc. Архіви та утиліта jar.

Розділ 4. Бібліотека колекцій**Тема 14. Структура колекцій (пакет java.util). Послідовні колекції.**

Поняття колекції. Інтерфейс List і його реалізації (ArrayList, LinkedList). Інтерфейс Set и его реализации (HashSet, TreeSet). 1. Доступ до колекції через ітератор. Порівняння послідовних колекцій.

Тема 15. Структура колекцій (пакет java.util). Асоціативні Колекції.

Ілюстрація використання асоціативних колекцій. Бінарні дерева та хешування. Приклади. Карти відображень. Компаратори. Клас Arrays.

Розділ 5. Потоки.**Тема 16. Програмування множинних потоків. Огляд курсу**

Багато поточне програмування. Модель потоків в Java. Головний потік, створення потоку. Створення множинних потоків, пріоритети потоків. Міжпоточні зв'язки.

Тема 17. Робота з потоками.

Синхронізація. Внутрішні блокування. Живучість (Liveness): взаємне блокування (Deadlock), голодування (starvation), активна локіровка (livelock). Захищені блокування (guarded blocks). Незмінні об'єкти (immutable objects).

Тема 18. Введення / виведення в Java.

Потоки введення / виведення. Клас File. Байтові потоки. Символьні потоки. Робота з файлами з довільним доступом. Серіалізація.

Розділ 6. AWT, Swing. Введення в проектування

Тема 19. Основи проектування.

Мова UML та case засоби. Діаграми UML. Варіанти використання і дійові особи.

Тема 20. Основи об'єктно-орієнтованого проектування.

Взаємодія об'єктів. Класи і пакети. Атрибути, операції,

Тема 21. Генерація програмного коду.

Підготовка до генерації програмного коду. Генерація програмного коду Java. Зворотне проектування програмного коду Java.

Тема 22. Основи AWT: робота з вікнами.

Основи віконної графіки. Робота з фреймової вікнами. Малювання простих об'єктів.

Тема 23. 8 Обробка подій: модель делегування подій.

Події та їх джерела. Інтерфейси слухачів. Класи-адаптери.

Тема 24. Основи AWT: робота з текстом.

Робота з кольором. Робота зі шрифтом. Клас FontMetrics. Основи AWT: елементи управління (Кнопки. Списки. Панелі меню), використання менеджерів компоновання (BorderLayout, FlowLayout, GridLayout, CardLayout).

Тема 25. Аплети.

Загальні відомості про аплетах. . Методи алетів, висновок графіки. Питання безпеки.

Тема 26. Робота з бібліотекою Swing.

Клас JFrame. Створення стандартних елементів управління Swing (кнопки і їх групи, радиокнопки, прапорці, списки, що випадають і ін.). Бібліотека Swing: каталог компонентів: текстові поля, бордюри, панелі скролінгов, списки що випадають та ін.

Тема 27. Робота з бібліотекою Swing.

Каталог компонентів: меню, спливаючі меню, вікна діалогу, файлові діалоги. Елементи GUI і пов'язані з ними події. Модель проектування MVC. Порівняння побудови програми на основі MVC та без нього на прикладі.

Тема 28. Огляд Java FX.

Огляд Java FX (технологія Java Beans, BDK, інтроспекція).

3. Структура навчальної дисципліни

Назви розділів і тем	Кількість годин					
	денна форма					
	усього	у тому числі				
л		п	лаб.	інд.	с. р.	
1	2	3	4	5	6	7
II семестр						
Розділ 1. Знайомство з мовою Java. Інкапсуляція.						
Тема 1. Огляд курсу.	10	2		2		6
Тема 2. Основні поняття мови Java.	10	2		2		6
Тема 3. Основні поняття мови Java.	11	2		2	1	6
Тема 4. Структурні типи даних.	11	2		2		7
Тема 5. Об'єктна модель Java.	13	2		2	1	8
Разом за розділом 1	55	10		10	2	33
Розділ 2. Спадкування та поліморфізм						
Тема 6. Повторне використання класів.	12	2		2		8
Тема 7. Об'єктна модель в Java.	12	2		2		8
Тема 8. Об'єктна модель в Java.	13	2		2	1	8
Разом за розділом 2	37	6		6	1	24
Розділ 3. Обробка виключень, внутрішні класи і огляд стандартних пакетів.						
Тема 9. Внутрішні класи.	12	2		2		8
Тема 10. Обробка виключень.	12	2		2		8
Тема 11. Пакет java.lang.	12	2		2		8
Тема 12. Сервісні класи пакету java.util.	12	2		2		8
Тема 13. Пакети і утиліти.	11	2		2	1	6
Разом за розділом 3	59	10		10	1	38
Розділ 4. Бібліотека колекцій						
Тема 14. Структура колекцій. Послідовні колекції.	12	2		2		8
Тема 15. Структура колекцій. Асоціативні Колекції.	17	4		4	1	8
Разом за розділом 4	29	6		6	1	16
Усього за II семестр	180	32		32	5	111
III семестр						
Розділ 5. Потоки.						
Тема 16. Програмування множинних потоків. Огляд курсу	8	2		2	3	1
Тема 17. Робота з потоками.	13	4		4	5	
Тема 18. Введення / виведення в Java.	13	4		4	5	
Разом за розділом 5	34	10		10	13	1
Розділ 6. AWT, Swing. Введення в проектування						
Тема 19. Основи проектування.	7	2		2	2	1
Тема 20. Основи об'єктно-орієнтованого проектування.	14	4		4	5	1
Тема 21. Генерація програмного коду.	7	2		2	2	1
Тема 22. Основи AWT: робота з вікнами.	7	2		2	2	1
Тема 23. Обробка подій: модель делегування подій.	9	2		2	4	1
Тема 24. Основи AWT: робота з текстом.	7	2		2	2	1
Тема 25. Аплети.	9	2		2	4	1
Тема 26. Робота з бібліотекою Swing.	10	2		2	5	1
Тема 27. Робота з бібліотекою Swing.	9	2		2	4	1
Тема 28. Обзор Java FX.	7	2		2	2	1
Разом за розділом 6	86	22		22	32	10
Усього за III семестр	120	32		32	45	11
Усього годин	300	64		64	50	122

4. Теми практичних (лабораторних) занять

№ з/п	Назва теми	Кількість годин
<i>II семестр</i>		
Розділ 1. Інкапсуляція		
1	Введення-виведення, цикли, масиви	4
2	Інкапсуляція. Масиви об'єктів.	6
Розділ 2. Спадкування та поліморфізм		
3	Відкрите спадкування. Абстрактні класи. Поліморфізм	2
4	Інтерфейси.	4
Розділ 3. Обработка исключений, внутренние классы и обзор стандартных пакетов.		
5	Внутрішні класи	2
6	Обробка виключень.	2
7	Пакет java.lang. Пакет java.util.	6
Розділ 4. Бібліотека колекцій		
8	Послідовні колекції	2
9	Асоціативні колекції	4
<i>III семестр</i>		
Розділ 5. Потоки.		
10	Багатопотокове програмування. Робота з декількома потоками.	4
11	Потоки вводу. вивода	4
12	Серілізація.	4
Розділ 6. AWT, Swing. Введення в проектування		
13	Розробка програми з використанням AWT	4
14	Розробка програми з використанням Swing	8
15	Розробка аплету	8
	Разом	64

5. Завдання для самостійної роботи

№ з/п	Види, зміст самостійної роботи	Кількість годин
<i>II семестр</i>		
Розділ 1. Інкапсуляція		
1	Введення-виведення, цикли, масиви. Самостійне опрацювання лекційного матеріалу Додатково прочитати у Хорстманна[1] (глава 3) або Шилдта[3] (глави 3-5) про змінні, типи даних, константи та розібрати приклади коду. Розібратись з вводом з клавіатури за допомогою класу Scanner. Контроль: опитування під час практичного заняття, опитування під час здачі лабораторної роботи № 1	12
2	Інкапсуляція. Самостійне опрацювання лекційного матеріалу. Хорстманна[1] (глава 4), Шилдт[3] (глава 6). Контроль: опитування під час практичного заняття, опитування під час здачі лабораторної роботи № 2	12
3	Розробка корисного класу вектор. Самостійне опрацювання лекційного матеріалу. Хорстманна[1], глава 4, Шилдт[3] (глава 7). Контроль: опитування під час практичного заняття.	13
Розділ 2. Спадкування та поліморфізм		
4	Відкрите спадкування. Самостійне опрацювання лекційного матеріалу. Хорстманн[1], глава 5, Шилдт[3] (глава 8). Контроль: опитування під час практичного заняття, опитування під час прийому лабораторної роботи № 3. Підготовка до контрольної роботи	8

№ з/п	Види, зміст самостійної роботи	Кількість годин
5	Порівняння спадкування реалізації та композиції. Самостійне опрацювання лекційного матеріалу. Хорстманн[1], глава 5. Контроль: опитування під час практичного заняття, опитування під час прийому лабораторної роботи № 4	8
6	Поліморфізм. Інтерфейси. Самостійне опрацювання лекційного матеріалу. Хорстманн[1], глава 6 (інтерфейси), Шилдт[3] (глава 9). Контроль: опитування під час практичного заняття, Підготовка до контрольної роботи.	9
Розділ 3. Обработка исключений, внутренние классы и обзор стандартных пакетов.		
7	Внутрішні класи. Самостійне опрацювання лекційного матеріалу. Хорстманн[1], глава 6 (внутр.класи). Контроль: опитування під час практичного заняття, опитування під час прийому лабораторної роботи № 5	13
8	Обробка виключень. Самостійне опрацювання лекційного матеріалу. Хорстманн[1], глава 7, Шилдт[3] (глава 10). Контроль: опитування під час практичного заняття, опитування під час прийому лабораторної роботи № 6	13
9	Пакет java.lang. Пакет java.util. Самостійне опрацювання лекційного матеріалу. Шилдт[3] (глави 16, 18). Контроль: опитування під час практичного заняття, опитування під час прийому лабораторної роботи № 7	13
Розділ 4. Бібліотека колекцій		
10	Послідовні колекції Самостійне опрацювання лекційного матеріалу. Хорстманн[1], глава 9, Шилдт[3] (глави 17). Контроль: опитування під час практичного заняття, опитування під час прийому лабораторної роботи № 8	8
11	Асоціативні колекції. Самостійне опрацювання лекційного матеріалу. Контроль: опитування під час практичного заняття, опитування під час прийому лабораторної роботи № 9.	9
III семестр		
Розділ 5. Потoki.		
10	Багатопотокове програмування. Робота з декількома потоками. Самостійне опрацювання лекційного матеріалу. Шилдт[3] (глава 11, 27). Контроль: опитування під час практичного заняття, опитування під час прийому лабораторної роботи № 1 . Підготовка до контрольної роботи	1
11	Потоки вводу.вивода. Робота з декількома потоками. Самостійне опрацювання лекційного матеріалу. Шилдт[3] (глава 19). Контроль: опитування під час практичного заняття	
12	Серілізація. Самостійне опрацювання лекційного матеріалу. Контроль: опитування під час практичного заняття, опитування під час прийому лабораторної роботи № 2	
Розділ 6. AWT, Swing. Введення в проектування		
13	Розробка програми з використанням AWT. Самостійне опрацювання лекційного матеріалу. Шилдт[3] (глави 23,24,25). Контроль: опитування під час практичного заняття, , опитування під час прийому лабораторних робіт № 3,4.	4
14	Розробка програми з використанням Swing. Самостійне опрацювання лекційного матеріалу. Шилдт[3] (глави 26,30). Контроль: опитування під час практичного заняття, , опитування під час прийому лабораторних робіт № 5,6.	3
15	Розробка аплету. Самостійне опрацювання лекційного матеріалу. Шилдт[3] (глави 22). Контроль: опитування під час практичного заняття, опитування під час прийому лабораторної роботи № 7.	3
	Курсовой проект	40
	Разом	176

6. Індивідуальні завдання

Індивідуальна робота складається з виконання лабораторних робіт з використанням мови Java згідно методичним вказівкам та курсової роботи.

	Лабораторні завдання
2 семестр	9
3 семестр	7

Курсова робота. Як наукове та навчально-дослідне завдання кожен студент має виконати курсову роботу. В ході роботи над проектом студент повинен показати розуміння основ ООП та основ ООП проектування, графічного інтерфейсу, концепції MVC, роботи з файлами. Студент повинен розробити програму, пояснювальну записку до неї, створити презентацію та захистити свою роботу перед комісією та студентами. Кожному студенту дається можливість самостійно вибрати тему для виконання. Студент повинен обговорити тему роботи з викладачем та затвердити її.

В якості теми роботи може бути, наприклад

- 1) гра (сапер, змія, морський бій і т.п.)
- 2) термінал для роботи з «квазіБД» (реєстратура, бібліотека і т.п.) (зберігання даних в файлах)
- 3) тренажер (запам'ятовування іноземних слів, тестування з якогось предмету і т.п.)

7. Методи контролю

У другому семестру передбачений захист лабораторних робіт та індивідуальних завдань перед викладачем з одночасним їх тестуванням. Залік в кінці семестру складається з теста та балів за лабораторні та індивідуальні завдання.

У третьому семестрі передбачений захист лабораторних робіт перед викладачем, індивідуальної роботи, а також спільний проект, при виконанні якого прогрес контролює викладач. Результати роботи над проектом захищаються. Програмні продукти, розроблені відповідно темам курсових робіт, на CD диску та пояснювальні записки у вигляді паперових звітів зберігаються на кафедрі.

Екзамен в кінці семестра складається з теста та практичного завдання.

8. Схема нарахування балів

2 семестр.

Поточний контроль, самостійна робота, індивідуальні завдання										Залікова робота	Сума	
ЛР 1	ЛР 2	ЛР 3	ЛР 4	ЛР 5	ЛР 6	ЛР 7	ЛР 8	ЛР 9	Контрольна робота, передбачена навчальним планом			Разом
5	5	5	5	5	5	5	5	5	15	60	40	100

3 семестр.

Поточний контроль, самостійна робота, індивідуальні завдання									Екзамен	Сума	
ЛР 1	ЛР 2	ЛР 3	ЛР 4	ЛР 5	ЛР 6	ЛР 7	Індивідуальне завдання (курсів проект)	Контрольна робота, передбачена навчальним планом			Разом
5	5	5	5	5	3	2	20	10	60	40	100

Шкала оцінювання

Сума балів за всі види навчальної діяльності протягом семестру	Оцінка	
	для чотирирівневої шкали оцінювання	для дворівневої шкали оцінювання
90 – 100	відмінно	зараховано
70-89	добре	
50-69	задовільно	
1-49	незадовільно	не зараховано

Критерії оцінювання

Оцінка в балах	Оцінка за національною шкалою	Пояснення
90 – 100	Відмінно	Теоретичний зміст курсу освоєний цілком, необхідні практичні навички роботи з освоєним матеріалом сформовані, всі навчальні завдання, які передбачені програмою навчання виконані повному обсязі, відмінна робота без помилок або з однією незначною помилкою.
70 – 89	Добре	Теоретичний зміст курсу освоєний цілком, практичні навички роботи з освоєним матеріалом в основному сформовані, всі навчальні завдання, які передбачені програмою навчання виконані, якість виконання жодного з них не оцінено мінімальним числом балів, деякі види завдань виконані з помилками, робота з декількома незначними помилками, або з однією – двома значними помилками.
50 – 69	Задовільно	Теоретичний зміст курсу освоєний не повністю, але прогалини не носять істотного характеру, необхідні практичні навички роботи з освоєним матеріалом в основному сформовані, більшість передбачених програмою навчання навчальних завдань виконано, деякі з виконаних завдань, містять помилки, робота з трьома значними помилками
1–49	Незадовільно	Теоретичний зміст курсу не освоєно, необхідні практичні навички роботи не сформовані, всі виконані навчальні завдання містять грубі помилки, додаткова самостійна робота над матеріалом курсу не приведе до значимого підвищення якості виконання навчальних завдань, робота, що потребує повної переробки

9. Рекомендована література

Основна література

1. Хорстманн. Java 2. Том 1. Основи. 2007
2. Хорстманн. Java 2. Том 2. Тонкощі програмування. 2007
3. Шилдт Г. Java. Повне керівництво. 8-е видання. 2012

Допоміжна література

1. Еккель. Філософія Java

10. Посилання на інформаційні ресурси в Інтернеті, відео-лекції, інше методичне забезпечення

1. Литвинов Д.Н.: [Електроний ресурс]// Литвинов Д.Н., 2012-2015. URL:
<http://www.litvinov.in.ua>
2. Java tutorials: [Електроний ресурс]// Oracle and/or its affiliates, 1994-2017. URL:
<http://docs.oracle.com/javase/tutorial/reallybigindex.html>
3. Java™ Platform, Standard Edition 7 API Specification: [Електроний ресурс]// Oracle and/or its affiliates, 1993- 2017. URL:<http://docs.oracle.com/javase/7/docs/api/>