


ЛЬВІВСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ БЕЗПЕКИ ЖИТТЄДІЯЛЬНОСТІ

**КАФЕДРА УПРАВЛІННЯ ПРОЕКТАМИ, ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ ТА
ТЕЛЕКОМУНІКАЦІЙ**

ЗАТВЕРДЖУЮ

Голова Вченої ради

Навчально-наукового інституту
цивільного захисту

 Василь ПОПОВИЧ
" 04 " вересня 2020р.

ОК 2.8 ОБ'ЄКТНО-ОРІЄНТОВАНЕ ПРОГРАМУВАННЯ

ПРОГРАМА

навчальної нормативної дисципліни

підготовки бакалавра

спеціальності: 122 Комп'ютерні науки

за освітньою програмою: Комп'ютерні науки

Розробники програми:

Олександр Придатко, начальник кафедри управління проектами, інформаційних технологій та телекомунікацій, канд. тех. наук, доцент;

Богдан Кокотко, Software engineer, Scanmarket

Рецензент: Дмитро Пелешко, професор кафедри систем штучного інтелекту Національного університету «Львівська політехніка», професор, д-р. тех. наук

Програму рекомендовано кафедрою управління проектами, інформаційних технологій та телекомунікацій

Протокол від “27” серпня 2020 року № 1

Начальник (завідувач) кафедри управління проектами, інформаційних технологій та телекомунікацій


(підпис)

Олександр ПРИДАТКО
(ім'я та прізвище)

Схвалено Вченою радою навчально-наукового інституту цивільного захисту

Протокол від “04” вересня 2020 року № 1

ВСТУП

Програма вивчення нормативної навчальної дисципліни “Об’єктно-орієнтоване програмування” складена відповідно до освітньо-професійної програми підготовки бакалаврів зі спеціальності 122 “Комп’ютерні науки”.

Предметом вивчення навчального курсу є інструментальні засоби та технології створення програмних систем за об’єктно-орієнтованою парадигмою програмування. В ході курсу здобувачі вивчають тонкощі застосування об’єктно-орієнтованої мови програмування Java та середовищ розробки Eclipse і IntelliJ IDEA. За результатами вивчення курсу здобувачі готують індивідуальні (групові) проекти у вигляді реалізації програмної системи за об’єктно-орієнтованим принципом.

Міждисциплінарні зв’язки. Курс є ключовим в програмі підготовки бакалавра за спеціальністю 122 «Комп’ютерні науки» та являється основою для подальшого навчання та вивчення таких дисциплін як: «Клієнт-серверне програмування», «Об’єктне моделювання програмних систем», «Технології розподілених систем та паралельних обчислень», «Операційні системи та системне програмування», «Програмування для мобільних платформ», «Людино-машинна взаємодія» тощо. Сам курс опирається на фундаментальний курс «Основи програмування».

Програма навчальної дисципліни складається з таких **змістових модулів та тем:**

Змістовий модуль 1. Загальні принципи об’єктно-орієнтованого програмування

- Тема 1. Класи та методи.
- Тема 2. Робота з методами (поліморфізм)
- Тема 3. Робота з членами класу (інкапсуляція)
- Тема 4. Робота з класами (побудова ієрархії класів)
- Тема 5. Сумісне застосування парадигм ООП при створенні елементів програмних систем
- Тема 6. Пакети та інтерфейси (поглиблене вивчення)
- Тема 7. Опрацювання символічних стрічок
- Тема 8. Базові поняття UML. Діаграми класів

Змістовий модуль 2. Особливості об’єктно-орієнтованого програмування мовою Java

- Тема 9. Видача завдань для виконання курсової роботи
- Тема 10. Опрацювання винятків
- Тема 11. Багатопотокове програмування
- Тема 12. Потoki введення-виведення (пакет java.io)
- Тема 13. Лямбда-вирази
- Тема 14. Бібліотека Java: пакет java.lang
- Тема 15. Бібліотека Java: пакет java.util (колекції)
- Тема 16. Робота в мережі

1. МЕТА ТА ЗАВДАННЯ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

1.1. **Метою** навчальної дисципліни “Об’єктно-орієнтоване програмування” є поглиблення знань студентів з технології об’єктно-орієнтованого програмування, детальне вивчення етапів розробки програмного забезпечення за об’єктно-орієнтованою парадигмою, принципами побудови програмних систем за модульним принципом, а також написання свого першого застосунку в рамках виконання курсової роботи.

1.2. **Основними завданнями** вивчення дисципліни “Об’єктно-орієнтоване програмування” є здобуття поглиблених знань з об’єктно-орієнтованого моделювання програмних систем, їх прототипування з дотриманням основних парадигм ООП використовуючи як засіб реалізації об’єктно-орієнтовану мову програмування Java.

1.3. Програмні результати навчання:

- Застосовувати знання основних форм і законів абстрактно-логічного мислення, основ методології наукового пізнання, форм і методів вилучення, аналізу, обробки та синтезу інформації в предметній області комп’ютерних наук
- Розробляти програмні моделі предметних середовищ, вибирати парадигму програмування з позицій зручності та якості застосування для реалізації методів та алгоритмів розв’язання задач в галузі комп’ютерних наук
- Застосовувати знання методології та CASE-засобів проектування складних систем, методів структурного аналізу систем, об’єктно-орієнтованої методології проектування при розробці і дослідженні функціональних моделей організаційно-економічних і виробничо-технічних систем.

На вивчення навчальної дисципліни відводиться 195 годин(и)/ 6,5 кредитів ECTS.

2. ІНФОРМАЦІЙНИЙ ОБСЯГ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

ЗМІСТОВИЙ МОДУЛЬ 1.

ЗАГАЛЬНІ ПРИНЦИПИ ОБ'ЄКТНО-ОРІЄНТОВАНОГО ПРОГРАМУВАННЯ

Тема 1. Класи та методи.

Введення в класи. Структура класу. Введення в методи. Типізовані методи, void-методи, методи, що приймають параметри. Конструктори. Пусті конструктори, генерування конструкторів. Створення власних класів з унікальною логікою. Застосування конструкторів. Створення власних методів з унікальною логікою.

Тема 2. Робота з методами (поліморфізм)

Перевизначення методів. Перевизначення конструкторів. Перевантаження методів. Робота з об'єктами в методі. Передавання об'єктів в якості параметра методу. Застосування поліморфізму при створенні модулів програмних систем. Рекурсивні методи.

Тема 3. Робота з членами класу (інкапсуляція)

Інкапсуляція. Організація доступу до інкапсульованих членів класу. Управління доступом в ієрархії класів. Робота з інкапсульованими змінними та методами. Використання методів get() та set().

Тема 4. Робота з класами (побудова ієрархії класів)

Вкладені та внутрішні класи. Наслідування класів. Ключове слово super та його застосування. Побудова однорівневої ієрархії класів. Побудова багаторівневої ієрархії. Створення та використання вкладених класів.

Тема 5. Сумісне застосування парадигм ООП при створенні елементів програмних систем

Практика сумісного застосування наслідування та поліморфізму. Перевизначення та перевантаження методів при наслідуванні. Ключове слово final в перевизначенні та наслідуванні. Реалізація абстрактних класів та методів. Динамічна диспетчеризація методів.

Тема 6. Пакети та інтерфейси (поглиблене вивчення)

Пакети. Захист доступу. Імпорт пакетів. Інтерфейси та їх застосування. Вкладені інтерфейси. Змінні в інтерфейсах. Розширення інтерфейсів. Методи за замовчуванням. Застосування статичних методів в інтерфейсах.

Тема 7. Опрацювання символьних стрічок

Конструктори символьних стрічок. Спеціальні стрічкові операції. Метод toString(). Вилучення символів. Метод equals () та compareTo (). Пошук в символьних стрічках. Взаємодія символьних стрічок. Перетворення даних методом valueOf (). З'єднання стрічок та застосування інших методів класу String.

Тема 8. Базові поняття UML. Діаграми класів

Вступ у процес моделювання. Моделювання класів та об'єктів. Методологія об'єктно-орієнтованого моделювання. Загальна характеристика UML. Класифікація UML-діаграм. Правила і загальні механізми мови UML. Діагра-

ма класів та її застосування. Інструменти побудови діаграми класів. Побудова діаграми класів засобами бібліотеки Visio та ObjectAid UML Explorer.

ЗМІСТОВИЙ МОДУЛЬ 2.

ОСОБЛИВОСТІ ОБ'ЄКТНО-ОРІЄНТОВАНОГО ПРОГРАМУВАННЯ МОВОЮ JAVA

Тема 9. Видача завдань для виконання курсової роботи

Видача індивідуальних або групових завдань для виконання курсової роботи. Написання специфікації. Створення алгоритму або псевдокоду роботи програмної системи або її елементів. Побудова діаграми класів програмної системи засобами UML. Імплементация програмного коду. Перевірка працездатності та стійкості програмного коду. Написання коментарів до програмного коду.

Тема 10. Опрацювання винятків

Основи опрацювання винятків. Використання блоків операторів try та catch. Застосування операторів throw, throws та finally. Підкласи винятків, що не перевіряються (не оголошуються в методі). Підкласи винятків, що перевіряються (потрібно оголошувати в методі). Перехоплення та опрацювання винятків. Створення власних підкласів винятків.

Тема 11. Багатопотокове програмування

Вступ в багатопотокове програмування. Створення потоків виконання. Керування потоками виконання та їх синхронізація. Спільна взаємодія потоків виконання.

Тема 12. Потоки введення-виведення (пакет java.io)

Введення та виведення даних. Зчитування даних з консолі. Запис даних. Клас PrintWriter. Зчитування та запис даних у файл. Автоматичне закриття файлу. Класи та інтерфейси для вводу-виводу. Клас File. Клас Console. Потоки введення-виведення байтів. Потоки введення-виведення символів. Сериалізація.

Тема 13. Лямбда-вирази

Введення в лямбда-вирази. Блочні лямбда-вирази. Передача лямбда-виразів в кості аргументів. Лямбда-вирази та винятки. Лямбда вирази та захоплення змінних. Посилання на методи. Посилання на конструктори.

Тема 14. Бібліотека Java: пакет java.lang

Оболонки примітивних типів даних. Класи Void, Process, Runtime, System, Object, Class, Math та інші класи бібліотеки. Застосування класів пакету java.lang. Інтерфейси пакету java.lang та їх застосування.

Тема 15. Бібліотека Java: пакет java.util (колекції)

Огляд колекцій. Інтерфейси колекцій. Класи колекцій. Доступ до колекцій через ітератор. Компаратор та його застосування. Алгоритми колекцій.

Тема 16. Робота в мережі

Основи роботи в мережі. Мережеві класи та інтерфейси. Класи InetAddress, URL, URLConnection, HttpURLConnection, URI. Клієнтські сокети за протоколом TCP/IP. Серверні сокети за протоколом TCP/IP. Cookie-файли. Дейтаграми.

3. РЕКОМЕНДОВАНА ЛІТЕРАТУРА

Базова

1. **Java 8. The Complete Reference.** 9 edition / Herbert Schildt. – New York : Oracle Press, 2015. – 1274 с.
2. **Spring 5 для професіоналів** / Кларенс Хо, Кріс Шефер, Роб Харроп, Юліана Козміна. – Наука і техінка, 2020. – 1120 с.
3. **Об'єктно-орієнтоване моделювання програмних систем** : навчальний посібник. – Львів: Видавничий центр ЛНУ імені Івана Франка, 2007. – 108 с.
4. **Java: керівництво для початківців** / Шилдт Г. – К. : «Діалектик», 2020. – 816 с.
5. **Head First Java** / Kathy Sierra, Bert Bates. – «O'REILLY», 2012. – 678 с.
6. **Дизайн-патерни – просто, як двері** : підручник / Андрій Будай : «Developer's SUCCESS», 2012. – 90 с.
7. **Філософія Java.** 4-е вид. : пер. з англ. / Брюс Еккель : «Пітер», 2009. – 640 с.
8. **Лямбда-вирази в Java 8. Функціональне програмування - в маси** : пер. з англ. А. А. Слінкіна / Річард Уорбертон. – М. : ДМК Прес, 2014. – 192 с.

Допоміжна

1. **Придатко О. В.** Інтеграція новаційного методу мобільного навчання в освітні проекти підготовки розробників програмного забезпечення / О. В. Придатко, В. В. Придатко, Ю. О. Борзов, В. Є. Дзень // Вісник ЛДУБЖД: Зб. наук. праць. Львів: ЛДУ БЖД, 2018. – №18. – С.70-80.
2. **Martyn Ye.** Software for Shelter's Fire Safety and Comfort Levels Evaluation / Martyn Ye., Smotr O., Burak N., Prydatko O., Malets I. // Communications in Computer and Information Science, Springer, Cham. – Vol. 1158, 2020. pp. 457-469 https://doi.org/10.1007/978-3-030-61656-4_31
3. **Придатко О. В.** Адаптивна інформаційно-довідкова система "UniBell" як складова частина проекту "Smart-університет" / О. В. Придатко, Н. Є. Бурак, В. Є. Дзень, М. С. Кунинець // Науковий вісник НЛТУ України : Зб. наук. праць. Львів: НЛТУ, 2020. - т. 30, № 5 – С. 113–121.
4. **Андрушко О. А.** Аналіз процесів використання Docker для побудови мікросервісів / О. А. Андрушко, Ю. О. Борзов, І. О. Малець, О. В. Придатко // Науковий вісник НЛТУ України: Зб. нак.праць. Львів: НЛТУ, 2017. - №9(27) – С.95-98.
5. **Prydatko O.** Informational System of Project Management in the Areas of Regional Security Systems' Development / O. Prydatko, O. Smotr, Yu. Borzov, I. Solotvinskyi, O. Didyk // 2018 IEEE Second Conference on Data Stream Mining & Processing. Lviv, 2018. – №2 – 187-192.
6. **Concurrent Programming in Java™: Design Principles and Patterns,** Second Edition : book / Doug Lea : «Addison Wesley», 1999. – 432 p.
7. **Основи програмування на Java** : методичні вказівки до лабораторного практикуму та самостійної роботи, частина перша / Бивойно П. Г.,

15. Інформаційні ресурси

1. **Освоюємо Java. Вікіпідручник** [Електронний ресурс]. – Доступний з https://uk.wikibooks.org/wiki/Освоюємо_Java
2. **Програмування на Java** [Електронний ресурс]. – <http://kostin.ws/java>
3. **Java. ProgLang** [Електронний ресурс]. – Доступний з <http://proglang.su/java>
4. **Java Course** [Електронний ресурс]. – Доступний з <http://java-course.ru/begin/operations>
5. Prometheus. Курс «**Основи програмування на Java**». [Електронний ресурс]. – Доступний з <https://edx.prometheus.org.ua/courses/KPI>
6. Prometheus. Курс «**Курс CS-50**». [Електронний ресурс]. – Доступний з <https://edx.prometheus.org.ua/courses/KPI>

4. КРИТЕРІЇ УСПІШНОСТІ НАВЧАННЯ ТА ФОРМА ПІДСУМКОВОГО КОНТРОЛЮ

При оцінюванні результатів навчання здобувачів освіти потрібно керуватися такими **критеріями успішності навчання**:

Бали	Оцінка	Критерії оцінювання
91–100	Відмінно	<p>Здобувач демонструє повні й вичерпні знання навчального матеріалу в обсязі, що відповідає робочій програмі дисципліни, правильно й обґрунтовано приймає необхідні рішення в різних нестандартних ситуаціях.</p> <p>Вміє реалізувати теоретичні положення дисципліни при розв’язуванні практичних завдань, може аналізувати і співставляти навчальний матеріал з даної та суміжних дисциплін. Знає сучасні технології та методи рішення прикладних завдань з дисципліни.</p> <p>За час навчання при проведенні практичних /лабораторних занять, виконанні індивідуальних завдань проявив вміння самостійно вирішувати поставлені завдання, активно включатись в дискусії, може відстоювати власну позицію в питаннях та рішеннях, що розглядаються.</p> <p>Зменшення 100-бальної оцінки може бути пов’язане з недостатнім розкриттям питань, що стосується дисципліни яка вивчається, але виходить за рамки об’єму матеріалу передбаченого робочою програмою, або здобувач проявляє невпевненість в тлумаченні теоретичних положень чи рішення складних практичних завдань.</p>
81–90	Добре	<p>Здобувач демонструє добрі та вичерпні знання, володіє матеріалом, що відповідає робочій програмі дисципліни, робить на основі здобутих знань аналіз можливих ситуацій та вміє засто-</p>

Бали	Оцінка	Критерії оцінювання
		<p>совувати теоретичні положення при рішенні практичних завдань, проте допускає окремі неточності. Вміє самостійно виправляти допущені помилки, кількість яких є незначною. Знає сучасні технології та методи рішення практичних завдань з дисципліни.</p> <p>За час навчання при проведенні практичних /лабораторних занять, виконанні індивідуальних завдань та поясненні прийнятих рішень, дає вичерпні пояснення.</p>
71–80	Добре	<p>Здобувач в загальному добре володіє матеріалом, знає основні положення матеріалу, що відповідають робочій програмі дисципліни, робить на їх основі аналіз можливих ситуацій та використовує для рішення характерних/типових прикладних завдань з дисципліни.</p> <p>Вміє пояснити основні положення виконаних завдань та давати правильні відповіді про зміну результату при заданій зміні вихідних параметрів. Помилки у відповідях / рішеннях / розрахунках не є системними.</p> <p>Розуміє основні положення, що мають визначальне значення для практичних / лабораторних занять, виконанні індивідуальних завдань в межах дисципліни.</p>
61–70	Задовільно	<p>Здобувач засвоїв основний теоретичний матеріал, передбачений робочою програмою дисципліни, та розуміє постановку стандартних практичних завдань, має пропозиції щодо напрямку їх вирішень.</p> <p>Розуміє основні положення, що є визначальними в курсі, може вирішувати завдання подібні тим, що розглядалися на заняттях, проте допускає значну кількість неточностей і помилок, усунути які здатен лише за допомогою викладача.</p>
51–60	Задовільно	<p>Здобувач володіє певними знаннями та основними положеннями, передбаченими робочою програмою дисципліни, на мінімально допустимому рівні для подальшого засвоєння результатів навчання в рамках освітньої програми. З використанням основних теоретичних положень здобувач з труднощами пояснює правила вирішення практичних завдань дисципліни.</p> <p>Виконання практичних / лабораторних завдань значно формалізовано: є відповідність алгоритму, проте відсутнє глибоке розуміння самої роботи.</p>
35–50	Незадовільно	<p>Здобувач може відтворити окремі фрагменти знань з курсу. Незважаючи на те, що програму навчальної дисципліни здобувач виконав, працював він пасивно, його відповіді під час практичних і лабораторних робіт та результати поточного контролю в більшості є невірними та/або необґрунтованими. Цілісність розуміння матеріалу з дисципліни у здобувача відсутні, що створює перепони для подальшого засвоєння результатів навчання в рамках освітньої програми.</p>
0–34	Незадовільно	<p>Здобувач повністю не виконав вимог робочої програми навчальної дисципліни. Його відповіді під час практичних і лабораторних робіт та результати поточного контролю є невірними</p>

Бали	Оцінка	Критерії оцінювання
		та/або необґрунтованими. Його знання на підсумкових етапах навчання є фрагментарними.

Критерії оцінювання курсової роботи подано у методичних вказівках її до виконання.

Формами підсумкового контролю є захист курсової роботи та екзамен.

5. ЗАСОБИ ДІАГНОСТИКИ УСПІШНОСТІ НАВЧАННЯ

Під час вивчення дисципліни передбачено індивідуальний поточний контроль, фронтальний контроль, контроль за виконанням самостійної роботи, самоконтроль та індивідуальний підсумковий контроль у формі екзамену (захисту курсової роботи). Поточний контроль здійснюється у формі виконання тестових завдань на базі платформи електронного освітнього середовища «Віртуальний університет». Самоконтроль організовано шляхом надання здобувачам освіти другої спроби для складання тестових завдань (можливість надолуження пройденого матеріалу та перевірки рівня його засвоєння). Фронтальний контроль передбачає проведення наскрізного тестування або усного опитування під час лекційних занять з метою визначення якості засвоєння нового матеріалу. Під час лабораторних занять або/та консультацій викладач здійснює контроль за самостійною роботою здобувачів освіти шляхом прийняття захисту лабораторних робіт, а також періодичного аналізу стану виконання курсової роботи (із наданням відповідних рекомендацій). Індивідуальний підсумковий контроль здійснюється у формі екзамену та захисту курсової роботи.

Усі форми контролю включено до 100-бальної шкали оцінювання. Оцінка із 100-бальної шкали в національну переводиться відповідно до діючого положення про освітній процес (91–100 – «відмінно», 71–90 – «добре», 51–70 – «задовільно», менше 51 – «незадовільно»).