


МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Львівський національний університет імені Івана Франка
Факультет електроніки та комп'ютерних технологій
Кафедра радіоелектронних і комп'ютерних систем

Затверджено

На засіданні
кафедри радіоелектронних і
комп'ютерних систем
факультету електроніки та комп'ютерних
технологій
Львівського національного університету
імені Івана Франка
(протокол № *1/4* від *28.08.* 2023 р.)

Завідувач кафедри:



Ігор ОЛЕНИЧ

Силабус з навчальної дисципліни
“Функціональне програмування”,
що викладається в межах ОПП
“ Високопродуктивний комп'ютинг ”
першого (бакалаврського) рівня вищої освіти для здобувачів з
спеціальності 121 – Інженерія програмного забезпечення

Львів 2023 р.

Назва дисципліни	Функціональне програмування
Адреса викладання дисципліни	Корпус факультету електроніки та комп'ютерних технологій, Львівський національний університет імені Івана Франка, вул. Драгоманова 50, м. Львів, 79005, вул. Ген. Тарнавського 107, м. Львів, 79011
Факультет та кафедра, за якою закріплена дисципліна	Факультет електроніки та комп'ютерних технологій Кафедра радіоелектронних і комп'ютерних систем
Галузь знань, шифр та назва спеціальності	12 – інформаційні технології 121 – Інженерія програмного забезпечення
Викладачі дисципліни	Флюнт О.Є., кандидат фіз.-мат. наук, доцент
Контактна інформація викладачів	Orest.Fliunt@lnu.edu.ua https://electronics.lnu.edu.ua/employee/fliunt-orest-yevhenovych/
Консультації з питань навчання по дисципліні відбуваються	Консультації в день проведення лекцій/лабораторних занять (за попередньою домовленістю): ауд. 101, корпус факультету електроніки та комп'ютерних технологій, вул. Драгоманова 50, м. Львів
Сторінка курсу	https://moodle.elct.lnu.edu.ua/course/view.php?id=61
Інформація про дисципліну	Дисципліна “Функціональне програмування” є вибірковою дисципліною з спеціальності 121 – Інженерія програмного забезпечення для освітньої програми “Високопродуктивний комп'ютинг”, яка викладається в 7-му семестрі в обсязі 6 кредитів (за Європейською Кредитно-Трансферною Системою ECTS).
Коротка анотація дисципліни	Курс розроблено для ознайомлення студентів із функціональним підходом до розробки прикладних програм, з існуючими технологіями функціонального програмування, окреслення ряду прикладних задач, які розв'язуються із використанням функціонального підходу, вивчення функціональної мови програмування LISP.
Мета та цілі дисципліни	<i>Метою</i> є сформуванню у студентів системні знання та вміння функціонального програмування на мові LISP та Python. Засвоєння студентами основних концепцій, принципів та понять сучасного функціонального програмування, що створюють основу теоретичних досліджень і практичних розробок в області функціональних мов програмування. <i>Цілі:</i> познайомити студентів з основами функціонального програмування. Дати уявлення про можливості функціонального підходу до задач інформатики, включаючи рішення задач штучного інтелекту й комп'ютерної лінгвістики.
Література для вивчення дисципліни	1. Заяць В.М. Логічне та функціональне програмування. Системний підхід: підручник / В.М. Заяць. – Рівне : НУВГП, 2018. – 422 с. 2. Месюра В.І. Математичні основи логічного програмування : навч. посіб. / В.І. Месюра, Н.В. Лисак, О.І. Суприган. – Вінниця : ВНТУ, 2013. – 94 с.

	<p>3. Шумейко О.О. Visual Prolog. Опануй на прикладах : навч. посіб. / О.О. Шумейко, В.М. Кнуренко. – Дніпропетровськ : Біла К.О., 2014. – 404 с.</p> <p>4. Eduardo Costa. Visual Prolog 7.3 for Tyros. / Eduardo Costa. – New York: Springer-Verlag, 2010. – 270p.</p> <p>5. Mark Lutz. Learning Python, 5th Edition - O'Reilly, 2013. 1643p.</p> <p>6. Allen C., Moronuki J. Haskell Programming from first principles. New York: Christopher Allen and Julie Moronuki, 2016. 1228 p.</p> <p>7. Програмування числових методів мовою PYTHON / А. Ю. Дорошенко [та ін.]; за ред. А. В. Анісімова. – ВПЦ "Київський університет", 2013. – 464 с.</p> <p>8. Програмування числових методів мовою Python : підруч. / А. В. Анісімов, А. Ю. Дорошенко, С. Д. Погорілий, Я. Ю. Дорогий ; за ред. А. В. Анісімова. – К. : Видавничо-поліграфічний центр "Київський університет", 2014. – 640 с.</p> <p>9. David Mertz Functional Programming in Python – O'Reilly Media, Inc., 2015 – 39 p.</p>
Обсяг курсу	Загальний обсяг: 180 годин. Аудиторних занять: 64 год., з них 32 год. лекційних та 32 годин лабораторних робіт. Самостійної роботи: 116 год.
Очікувані результати навчання	<p>У результаті вивчення даного курсу студент буде знати:</p> <ul style="list-style-type: none"> - теоретичні основи розробки та експлуатації програмних продуктів, що використовують засоби функціонального програмування; - сучасний стан та перспективи розвитку функціонального програмування; - синтаксис мови LISP; - особливості функціонування інтерпретатора мови LISP; - застосовувати функції вищих порядків; <p>вміти:</p> <ul style="list-style-type: none"> - створювати програмні продукти засобами функціонального програмування; - застосовувати методи функціонального програмування для розв'язання інтелектуальних задач; - використовувати функції мовою LISP, у тому числі функції вищих порядків; - застосовувати засоби функціонального програмування мови Python. <p>Після вивчення даного курсу «Функціональне програмування» здобувачі набувають таких Загальних(ЗК)/Фахових(ФК) компетентностей та Програмних результатів навчання (ПРН):</p> <p>ЗК01. Здатність до абстрактного мислення, аналізу та синтезу.</p> <p>ЗК02. Здатність застосовувати знання у практичних ситуаціях.</p> <p>ЗК03. Здатність спілкуватися державною мовою як усно, так і письмово.</p> <p>ЗК05. Здатність вчитися і оволодівати сучасними знаннями.</p> <p>ЗК06. Здатність до пошуку, оброблення та аналізу інформації з різних джерел.</p> <p>ЗК09. Прагнення до збереження навколишнього середовища.</p> <p>ЗК11. Здатність реалізувати свої права і обов'язки як члена суспільства, усвідомлювати цінності громадянського (вільного демократичного) суспільства та необхідність його сталого розвитку, верховенства права, прав і свобод людини і громадянина в Україні.</p> <p>ФК14. Здатність брати участь у проектуванні програмного забезпечення, включаючи проведення моделювання (формальний опис) його структури,</p>

	<p>поведінки та процесів функціонування.</p> <p>ФК15. Здатність розробляти архітектури, модулі та компоненти програмних систем.</p> <p>ФК17. Здатність дотримуватися специфікацій, стандартів, правил і рекомендацій в професійній галузі при реалізації процесів життєвого циклу.</p> <p>ФК18. Здатність аналізувати, вибирати і застосовувати методи і засоби для забезпечення інформаційної безпеки (в тому числі кібербезпеки).</p> <p>ФК23. Здатність реалізовувати фази та ітерації життєвого циклу програмних систем та інформаційних технологій на основі відповідних моделей і підходів розробки програмного забезпечення.</p> <p>ФК24. Здатність здійснювати процес інтеграції системи, застосовувати стандарти і процедури управління змінами для підтримки цілісності, загальної функціональності і надійності програмного забезпечення.</p> <p>ФК25. Здатність обґрунтовано обирати та освоювати інструментарій з розробки та супроводження програмного забезпечення.</p> <p>ФК26. Здатність до алгоритмічного та логічного мислення.</p> <p>ФК29. Здатність здійснювати розробку програмного забезпечення використовуючи різні методології та засоби програмування з метою забезпечення їх високої надійності та продуктивності в роботі.</p> <p>ПРН01. Аналізувати, цілеспрямовано шукати і вибирати необхідні для вирішення професійних завдань інформаційно-довідникові ресурси і знання з урахуванням сучасних досягнень науки і техніки.</p> <p>ПРН06. Вміння вибирати та використовувати відповідну задачі методологію створення програмного забезпечення.</p> <p>ПРН09. Знати та вміти використовувати методи та засоби збору, формулювання та аналізу вимог до програмного забезпечення.</p> <p>ПРН13. Знати і застосовувати методи розробки алгоритмів, конструювання програмного забезпечення та структур даних і знань.</p> <p>ПРН18. Знати та вміти застосовувати інформаційні технології обробки, зберігання та передачі даних.</p> <p>ПРН27. Знати основи інженерії даних і конструювання конвеєрів даних та вміти обирати оптимальні алгоритми і технології розробки інноваційних рішень, зокрема для вирішення задач наук про дані та вбудованих систем.</p>
Ключові слова	LISP, функціональне програмування, рекурсивні функції, функції вищих порядків, лямбда-числення.
Формат курсу	Очний
Теми	<p>Тема 1. Загальне уявлення про функціональне програмування та його застосування.</p> <p>Тема 2. Функціональне програмування в LISP: дані, примітивні функції, списки.</p> <p>Тема 3. Рекурсивне та ітеративне визначення функцій.</p> <p>Тема 4. Атоми, конси, списки. Розпізнавачі типу. Функції властивостей.</p> <p>Тема 5. Числові та логічні функції. Контрольні конструкції.</p> <p>Тема 6. Конструювання списків у LISP.</p> <p>Тема 7. Числові функції. Керуючі структури.</p> <p>Тема 8. Засоби виведення у LISP. Задачі породження комбінаторних об'єктів.</p> <p>Тема 9. Лямбда-числення та функції вищих порядків.</p> <p>Тема 10. Побудова бінарних дерев. Функції модифікації.</p> <p>Тема 11. Функції планування та їх застосування до груп послідовностей</p>

	<p>даних.</p> <p>Тема 12. Замикання засобами LISP. Фабрики функцій.</p> <p>Тема 13. Структури у мові LISP.</p> <p>Тема 14. Засоби функціонального програмування мови Python.</p> <p>Тема 15. Монади у функціональному програмуванні.</p> <p>Тема 16. Застосування діалектів мови LISP.</p>
Підсумковий контроль, форма	Залік у кінці семестру
Пререквізити	<p>Для вивчення даного курсу студентам потрібні базові знання з курсів:</p> <ul style="list-style-type: none"> - вища математика; - дискретна математика; - основи програмування; - алгоритми і структури даних; - об'єктно-орієнтоване програмування.
Навчальні методи та техніки, які будуть використовуватися під час викладання курсу	<p>Інформаційні методи (лекція, бесіда, ілюстрація, демонстрація);</p> <p>дедуктивні методи на основі узагальнень;</p> <p>евристичні методи (проблемна лекція);</p> <p>інтерактивні методи (дискусія).</p>
Необхідне обладнання	<p>Для проведення лекційних занять: комп'ютер (мінімальні характеристики: процесор Intel Core i3 (4 ядра/8 потоків), 8ГБ оперативної пам'яті, 50ГБ вільного місця на диску, доступ до мережі Internet, засоби мультимедіа (в т.ч. проектор).</p> <p>Для проведення лабораторних занять: Комп'ютер (мінімальні характеристики: процесор Intel Core i3 (4 ядра/8 потоків), 8ГБ оперативної пам'яті, 50ГБ вільного місця на диску. Необхідне програмне забезпечення включає в себе ОС Ubuntu 20 LTS, середовище розробки Thonny, інтерпретатор мови програмування Python 3.6, LISP.</p>
Критерії оцінювання (окремо для кожного виду навчальної діяльності)	<p>Оцінювання проводиться за 100-бальною шкалою. Бали нараховуються за наступним співвідношенням:</p> <ul style="list-style-type: none"> • Змістовий модуль 1: до 15 балів за виконання тестових завдань. • Змістовий модуль 2: до 15 балів за виконання тестових завдань. • Виконання лабораторних робіт: до 70 балів. <p>Підсумкова максимальна кількість балів 100. Звітність за курс – залік в кінці семестру.</p> <p>Змістовий модуль – 60 тестових завдань.</p> <p>Академічна доброчесність: Очікується, що роботи студентів будуть оригінальними дослідженнями чи міркуваннями. Списування та втручання в роботу інших студентів становлять, але не обмежують, приклади можливої академічної недоброчесності. Виявлення ознак академічної недоброчесності в написанні завдань є підставою для її незарахування</p>

викладачем, незалежно від масштабів плагіату чи обману.

Відвідування занять є важливою складовою навчання. Очікується, що всі студенти відвідають усі лекції та лабораторні заняття курсу. Студенти повинні інформувати викладача про неможливість відвідати заняття. У будь-якому випадку студенти зобов'язані дотримуватися термінів визначених для виконання всіх видів робіт, передбачених курсом.

Література. Уся література, яку студенти не зможуть знайти самостійно, буде надана викладачем виключно в освітніх цілях без права її передачі третім особам. Студенти заохочуються до використання також й іншої літератури та джерел, яких немає серед рекомендованих.

Політика виставлення балів. Враховуються бали, набрані при поточному контролі та бали підсумкового тестування. При цьому обов'язково враховуються присутність на заняттях та активність студента під час практичного заняття; недопустимість пропусків та запізнь на заняття; користування мобільним телефоном, планшетом чи іншими мобільними пристроями під час заняття в цілях не пов'язаних з навчанням; списування та плагіат; несвоєчасне виконання поставленого завдання і т. ін. Жодні форми порушення академічної доброчесності не толеруються.

Оцінювання лабораторних робіт (14 лабораторних робіт, максимальна кількість балів: 70) відбувається шляхом оцінки роботи студента під час захисту звіту по виконаній лабораторній роботі (0-5 балів за одну роботу). Бали оцінювання лабораторних робіт нараховуються за наступним співвідношенням:

- 5 – студент в повному обсязі володіє навчальним матеріалом, має повне розуміння розглянутої теми, надає правильні відповіді на запитання по темі, код програми функціонує відповідно до завдання;
- 4 – студент достатньо розуміє розглянутий матеріал та принципи написаного ним коду програми, присутні неточності та незначні помилки у відповідях на запитання по темі, код програми функціонує відповідно до завдання (або з несуттєвими недоліками);
- 3 – студент не досить добре розуміє розглянутий матеріал та написаний ним код програми, вагається та надає неточні/не конкретні відповіді на запитання по темі, код програми функціонує неточно, або з помірними недоліками;
- 2 – студент погано розуміє розглянутий матеріал та написаний ним код програми, студент в більшості надає помилкові відповіді на питання по темі, код програми функціонує з суттєвими недоліками;
- 1 - студент погано розуміє розглянутий матеріал та написаний ним код програми, код програми не функціонує належним чином;
- 0 - студент зовсім не засвоїв розглянутий матеріал, написаний ним код програми не відповідає темі/не функціонує взагалі.

Оцінювання змістових модулів (2 змістових модулів, 15 балів за кожний) — за результатами написаних студентом тестів.

Критерії оцінювання результатів неформальної освіти:
 Нарахування балів відбувається за написання студентом тез доповідей на конференціях, наукових статей, участь у діяльності наукових гуртків, участь у наукових семінарах та круглих столах, конкурсах, участь у заходах неформальної освіти за отримання сертифікатів про проходження навчання на різних освітніх платформах (Coursera, Prometheus тощо), курсах на провідних ІТ компаніях за тематикою навчальної дисципліни. Кількість балів визначається відсотком покриття результатів відповідної активності

	до вимог результатів навчання з навчальної дисципліни.
Питання до заліку чи екзамену.	Перелік питань на залік розміщений на веб-сторінці курсу.
Опитування	Анкету-оцінку з метою оцінювання якості курсу буде надано по завершенню курсу.

**Схема курсу “Функціональне програмування”
для студентів спеціальності 121 – Інженерія програмного забезпечення**

Тиж.	Тема, план, короткі тези	Форма діяльності (заняття)* *лекція, самостійна, дискусія, групова робота)	Література.** * Ресурси в інтернеті	Завдання, год	Термін виконання
1	Загальне уявлення про функціональне програмування та його застосування.	Лекція	[1], [2], [3], [4], Сайт курсу	2	кінець поточного тижня
1	Вступне заняття. Робота в середовищі Visual LISP.	Лабораторна робота	Сайт курсу	2	кінець поточного тижня
2	Функціональне програмування в LISP: дані, примітивні функції, списки.	Лекція	[1], [2], [3], [4], Сайт курсу	2	кінець поточного тижня
2	Основні конструкції мови LISP.	Лабораторна робота	Сайт курсу	2	кінець поточного тижня
3	Рекурсивне та ітеративне визначення функцій.	Лекція	[1], [2], [4] Сайт курсу	2	кінець поточного тижня
3	Створення програм для розв'язування логічних задач.	Лабораторна робота	Сайт курсу	2	кінець поточного тижня
4	Атоми, конси, списки. Розпізнавачі типу. Функції властивостей.	Лекція	[1], [2], [3], Сайт курсу	2	кінець поточного тижня
4	Застосування функцій на мові LISP.	Лабораторна робота	Сайт курсу	2	кінець поточного тижня
5	Числові та логічні функції. Контрольні конструкції.	Лекція	[1], [2], [6], Сайт курсу	2	кінець поточного тижня
5	Створення простих функцій для доступу до структурованих властивостей.	Лабораторна робота	Сайт курсу	2	кінець поточного тижня
6	Конструювання списків у LISP.	Лекція	[1], [2], [6], Сайт курсу	2	кінець поточного тижня
6	Структури розгалуження на мові LISP.	Лабораторна робота	Сайт курсу	2	кінець поточного тижня
7	Числові функції. Керуючі структури.	Лекція	[1], [2], [6], Сайт курсу	2	кінець поточного тижня

7	Числові функції в LISP. Циклічні обчислення.	Лабораторна робота	Сайт курсу	2	кінець поточного тижня
8	Засоби виведення у LISP. Задачі породження комбінаторних об'єктів.	Лекція	[1], [2], [6], Сайт курсу	2	кінець поточного тижня
8	Виведення даних у LISP.	Лабораторна робота	Сайт курсу	2	кінець поточного тижня
9	Лямбда-числення та функції вищих порядків.	Лекція	[1], [2], [6], [7] Сайт курсу	2	кінець поточного тижня
9	Застосування простих рекурсивних функцій в мові LISP.	Лабораторна робота	Сайт курсу	2	кінець поточного тижня
10	Побудова бінарних дерев. Функції модифікації.	Лекція	[1], [2], [5], [6], Сайт курсу	2	кінець поточного тижня
10	Створення рекурсивних функцій для роботи зі списками: лінійними та ієрархічними.	Лабораторна робота	Сайт курсу	2	кінець поточного тижня
11	Функції планування та їх застосування до груп послідовностей даних.	Лекція	[1], [2], [4], [7], Сайт курсу	2	кінець поточного тижня
11	Створення ієративних функцій для роботи зі списками: лінійними та ієрархічними.	Лабораторна робота	Сайт курсу	2	кінець поточного тижня
12	Замикання засобами LISP. Фабрики функцій.	Лекція	[1], [2], [3], [8], Сайт курсу	2	кінець поточного тижня
12	Функції та структури для моделювання багатовимірних масивів.	Лабораторна робота	Сайт курсу	2	кінець поточного тижня
13	Структури у мові LISP. Супутні функції. Наслідування структур.	Лекція	[1], [2], [3], Сайт курсу	2	кінець поточного тижня
13	Функції вищих порядків: числове інтегрування, табулювання функцій.	Лабораторна робота	Сайт курсу	2	кінець поточного тижня
14	Засоби функціонального програмування мови Python: лямбда-вирази, відображення функцій на послідовності.	Лекція	[5], [8], [9], Сайт курсу	2	кінець поточного тижня
14	Використання функцій планування мови LISP для отримання послідовності значень, для акумулювання результатів.	Лабораторна робота	Сайт курсу	2	кінець поточного тижня
15	Монади у функціональному програмуванні. Приклади реалізації та використання мовою Python. Бібліотека rmonad.	Лекція	[5], [7], [8], [9], Сайт курсу	2	кінець поточного тижня
15	Використання функціональних засобів мови Python.	Лабораторна робота	Сайт курсу	2	кінець поточного тижня
16	Застосування діалектів мови LISP.	Лекція	[1], [2], [3], Сайт курсу	2	кінець поточного тижня
16	Підсумкове заняття.	Лабораторна робота	Сайт курсу	2	кінець поточного тижня