



ЛОГІЧНЕ ПРОГРАМУВАННЯ

Робоча програма навчальної дисципліни (Силабус)

Реквізити навчальної дисципліни

Рівень вищої освіти	Перший (бакалаврський)
Галузь знань	12 Інформаційні технології
Спеціальність	121 Інженерія програмного забезпечення
Освітня програма	Інженерія програмного забезпечення інтелектуальних кібер-фізичних систем в енергетиці
Статус дисципліни	Вибіркова
Форма навчання	очна (денна)
Рік підготовки, семестр	3 курс, весняний семестр
Обсяг дисципліни	4,0 кредити ECTS /120 годин (36 годин лекцій, 18 годин практичних занять, 66 годин — самостійна робота студента)
Семестровий контроль/ контрольні заходи	Залік, МКР, представлення робіт комп'ютерного практикуму
Розклад занять	http://rozklad.kpi.ua
Мова викладання	Українська
Інформація про керівника курсу / викладачів	Лектор: к.т.н., доцент, Статівка Юрій Іванович, statyvka-yu@iit.kpi.ua Практичні заняття: к.т.н., доцент, Статівка Юрій Іванович, statyvka-yu@iit.kpi.ua
Розміщення курсу	https://ecampus.kpi.ua

Програма навчальної дисципліни

1. Опис навчальної дисципліни, її мета, предмет вивчення та результати навчання

Метою дисципліни є вивчення теоретичних основ та засобів створення програм в межах декларативної (логічної) парадигми; набуття студентами досвіду використання логічного програмування для розробки програмного забезпечення.

Предмет дисципліни — реалізація теоретичних основ логічного програмування у мові Пролог (Prolog); набуття студентами досвіду використання мови програмування Пролог для розробки програмного забезпечення.

Завдання. В результаті вивчення дисципліни у студентів повинні сформуватися наступні компетентності:

загальні:

- здатність до абстрактного мислення, аналізу та синтезу (ЗК 1);

- здатність застосовувати знання у практичних ситуаціях (ЗК 2);
- здатність спілкуватися державною мовою як усно, так і письмово (ЗК 3);
фахові:
- здатність накопичувати, обробляти та систематизувати професійні знання щодо створення тестування і супроводження програмного забезпечення та визнання важливості навчання протягом всього життя (ФК 10);
- здатність до алгоритмічного та логічного мислення (ФК 14).
Після засвоєння навчальної дисципліни студенти мають продемонструвати такі програмні результати навчання:
- Аналізувати, цілеспрямовано шукати і вибирати необхідні для вирішення професійних завдань інформаційно-довідникові ресурси і знання з урахуванням сучасних досягнень науки і техніки (ПРН 1);
- знати кодекс професійної етики, розуміти соціальну значимість та культурні аспекти інженерії програмного забезпечення і дотримуватись їх в професійній діяльності. (ПРН 2).

2. Пререквізити та постреквізити дисципліни (місце в структурно-логічній схемі навчання за відповідною освітньою програмою)

Пререквізити дисципліни. Знання, отримані при вивченні дисциплін: «Основи програмування», «Комп'ютерна дискретна математика», «Алгоритми та структури даних».

Постреквізити дисципліни. Компетенції, отримані студентами в процесі вивчення дисципліни можуть бути використані при виконанні дипломної роботи.

3. Зміст навчальної дисципліни

Розділ 1. Вступ до логічного програмування

Тема 1.1. Вступ до ЛП.

Тема 1.2. Інструментарій SWI-Prolog.

Тема 1.3. Модулі в Haskell.

Розділ 2. Основи мови SWI-Prolog

Тема 2.1. Основні елементи мови SWI-Prolog.

Тема 2.2. Арифметика в мові SWI-Prolog.

Тема 2.3. Узгодження цільових тверджень.

Тема 2.4. Рекурсія.

Тема 2.5. Структури.

Тема 2.6. Відсікання.

Тема 2.7. Введення-виведення в Пролозі.

Тема 2.8. Екстра-логічні засоби.

Тема 2.9. Деревя.

Тема 2.10. Графи.

Тема 2.11. Зовнішні інтерфейси SWI-Prolog.

Розділ 3. Розробка SWI-Prolog-застосунків

Тема 3.1. Граматика означених речень.

Тема 3.2. Доступ до баз даних.

Тема 3.3. Веб-застосунки.

Розділ 4. Концептуальні засади логічного програмування

Тема 4.1. Основи логічної парадигми.

4. Навчальні матеріали та ресурси

Основна література

1. Ivan Bratko Prolog Programming For Artificial Intelligence, 3rd Ed. / Ivan Bratko – Addison Wesley., 2000. – 678 p.
2. Leon Sterling, Ehud Shapiro The art of Prolog : advanced programming techniques / Leon Sterling, Ehud Shapiro; with a foreword by David H. D. Warren – MIT Press series in logic programming., 1999. – 550 p.

Інформаційні ресурси

3. Markus Triska. The Power of Prolog - <https://www.metalevel.at/prolog>
4. Офіційна сторінка книги “Real World Haskell” <http://book.realworldhaskell.org/>
5. Офіційна сторінка SWI-Prolog - <https://www.swi-prolog.org>
6. Документація SWI-Prolog, Посібник - https://www.swi-prolog.org/pldoc/doc_for?object=manual
7. SWI-Prolog-пакекти - https://www.swi-prolog.org/pldoc/doc_for?object=packages

Навчальний контент

5. Методика опанування навчальної дисципліни (освітнього компонента)

Лекційні заняття

Розділ 1. Вступ до логічного програмування

Лекція 1. Вступ до логічного програмування.

Порівняльний аналіз імперативної і декларативної парадигм програмування. Логічне програмування. Реалізації мови Пролог. SWI-Prolog. Мотиваційний приклад.

Лекція 2. Інструментарій SWI-Prolog.

SWI-Prolog: локальне встановлення; SWISH – онлайн-версія; Docker образ. Налаштування: командний рядок і GUI debugger. Редактор коду: визначений користувачем, Emacs, PseEmacs. Профілювання програм. Розширення VS Code: swi-prolog та інші. Пакети (Packages) SWI-Prolog.

Лекція 3. Модулі в Haskell.

Визначення модуля. Імпортування предикатів. Кероване автозавантаження модулів. Явне маніпулювання контекстом виклику. Взаємодія з модулями верхнього рівня. Оператори та модулі. Альтернативний інтерфейс імпорту/експорту. Динамічні модулі. Прозорі предикати: визначення та контекстний модуль. Властивості модуля.

Розділ 2. Основи мови SWI-Prolog

Лекція 4. Основні елементи мови SWI-Prolog.

Алфавіт мови. Терми. Види термів: константи, змінні, структури. Оператори. Властивості операторів (позиція, пріоритет, асоціативність). Інфіксні, префіксні та постфіксні оператори. Факти. Правила. Запис фактів і правил. Предикати. Цілі, кон'юнкція цілей. Загальна схема доведення цільового твердження.

Лекція 5. Арифметика в мові SWI-Prolog.

Вбудовані предикати для порівняння чисел. Обчислення арифметичних виразів. Предикат `is`. Приклади програм з виконанням арифметичних операцій.

Лекція 6. Узгодження цільових тверджень.

Доказ цільових тверджень при використанні механізму повернення. Правила встановлення відповідності. Недетермінізм першого і другого роду (`kind`). Поняття вільної та зв'язаної змінної. Приклади програм з використанням механізму повернення. Операційна і денотаційна моделі обчислення Prolog-програм.

Лекція 7. Рекурсія.

Рекурсивне подання даних і програм. Граничні умови і способи використання рекурсії. Кінцева рекурсія. Приклади програм з рекурсивними визначеннями.

Лекція 8. Структури.

Структури і дерева. Структурування даних, абстракція даних. Список як частковий вид структури. Форми запису списків. Робота зі списками. Сортування списків. Різницеві списки.

Лекція 9. Відсікання.

Відсікання і способи його використання. Причини використання відсікання. Предикат `cut`. Зелені та червоні відсікання. Загальні випадки використання відсікання. Проблеми, пов'язані з використанням відсікання.

Лекція 10. Введення-виведення в Пролозі.

Вбудовані предикати введення-виведення. Предикати для роботи з термами, символами, програмою. Приклади програм з використанням введення і виведення.

Лекція 11. Екстра-логічні засоби.

Динамічні предикати. Додавання і виключення тверджень, класифікація термів, зміна та аналіз тверджень, робота зі структурами довільного виду, вплив на процес повернення, оголошення операторів, обробка файлів, спостереження за виконанням Prolog-програми. Приклади використання вбудованих предикатів.

Лекція 12. Дерева.

Робота з деревами. Подання бінарних дерев і двійкових довідників в мові SWI-Prolog. 2-3-довідники. AVL-дерева. Приклади базових процедур роботи з деревами. Відображення дерев.

Лекція 13. Графи.

Представлення графів. Пошук шляху на графі. Побудова остовного дерева. Конвертація представлень графа. Приклади вирішення задач на графах. Оцінна функція. Пошук з пріоритетом. ТА/АБО-графи. Пошук в ТА/АБО-графах.

Лекція 14. Зовнішні інтерфейси SWI-Prolog.

Foreign Language Interface. Огляд інтерфейсів: C, C++, Java, JavaScript, Python, ODBC, RDF та інші. Інтеграція з Python: SWI-Prolog Python interface - виклик Python з SWI-Prolog та SWI-Prolog з Python.

Розділ 3. Розробка Prolog-застосунків

Лекція 15. Граматика означених речень.

Формальні граматика. DCG-граматика. Побудова DCG-граматик в SWI-Prolog. Приклади побудови граматик. Тестування та використання граматик.

Лекція 16. Доступ до баз даних.

Засоби зв'язку з базами даних. Пакети (надбудови, add-ons) proSQLite та db_facts.

Лекція 17. Веб-застосунки.

Пакет engines: розробка веб-застосунків на основі JavaScript, які взаємодіють з сервером Prolog. Віддалений виклик процедур через HTTP.

Розділ 4. Концептуальні засади логічного програмування

Лекція 18. Основи логічної парадигми.

Логіка предикатів першого порядку, Хорнівські диз'юнкти. Обчислювальна модель. Принцип Робінсона. Абстрактна машина Уорена.

Практичні заняття

Практичне заняття 1. Інструментарій і перший сеанс.

Середовище для роботи з SWI-Prolog. Робота в SWI-Prolog. Налаштування, профілювання, трасування. Факти, правила, запити.

Практичне заняття 2. Прості програми.

Зіставлення зі зразком (паттерном). Зразки на списках. Рекурсія.

Практичне заняття 3. Уніфікація.

Представлення роботи комп'ютерного практикуму № 1 (КП 1).

Практичне заняття 4. Робота зі структурами.

Представлення **КП 2**.

Практичне заняття 5. Екстра-логічні засоби.

Організація введення/виведення. Конструювання і аналіз предикатів.

Практичне заняття 6. Обробка нетривіальних структур.

Представлення **КП 3**.

Практичне заняття 7. SWI-Prolog Python interface.

Програмування з використанням двоспрямованого інтерфейсу.

Практичне заняття 8. SWI-Prolog-застосунки.
Представлення **КП 4**.

Практичне заняття 9. Підсумкове заняття.

Перелік робіт комп'ютерного практикуму

1. КП 1. Базові засоби мови Пролог.
2. КП 2. Структури і списки.
3. КП 3. Нетривіальні структури: пошук і трансформація.
4. КП 4. Розробка SWI-Prolog-застосунків.

Контрольна робота

Метою контрольної роботи є закріплення та перевірка теоретичних знань із освітнього компонента, набуття студентами практичних навичок самостійного вирішення задач.

Модульна контрольна робота (МКР) виконується на шістнадцятому тижні семестру і проводиться у формі тестування.

Самостійна робота студента

Приблизний розподіл часу СРС

№ з/п	Вид самостійної роботи	Кількість	Кількість годин СРС
1	Виконання робіт комп'ютерного практикуму	4	24
2	Підготовка до практичних занять	9	36
3	Підготовка до МКР	1	6
Разом			66

Політика та контроль

6. Політика навчальної дисципліни (освітнього компонента)

Система вимог при вивченні дисципліни:

- правила відвідування занять: заборонено оцінювати присутність або відсутність здобувача на аудиторному занятті, в тому числі нараховувати заохочувальні або штрафні бали. Відповідно до РСО даної дисципліни бали нараховують за відповідні види навчальної активності на лекційних та практичних заняттях.
- правила поведінки на заняттях: студент має можливість отримувати бали за відповідні види навчальної активності на лекційних та практичних заняттях, передбачені РСО дисципліни. Використання засобів зв'язку для пошуку інформації на гугл-диску викладача, в інтернеті, в дистанційному курсі на платформі Сікорський здійснюється за умови вказівки викладача;

- політика дедлайнів та перескладань: якщо студент не проходив або не з'явився на МКР (без поважної причини), його результат оцінюється у 0 балів. Перескладання результатів МКР не передбачено;
- політика щодо академічної доброчесності: Кодекс честі Національного технічного університету України «Київський політехнічний інститут» <https://kpi.ua/files/honorcode.pdf> встановлює загальні моральні принципи, правила етичної поведінки осіб та передбачає політику академічної доброчесності для осіб, що працюють і навчаються в університеті, якими вони мають керуватись у своїй діяльності, в тому числі при вивченні та складанні контрольних заходів з дисципліни «Логічне програмування»;
- при використанні цифрових засобів зв'язку з викладачем (мобільний зв'язок, електронна пошта, переписка на форумах та у соцмережах тощо) необхідно дотримуватись загальноприйнятих етичних норм, зокрема бути ввічливим та обмежувати спілкування робочим часом викладача.

7. Види контролю та рейтингова система оцінювання результатів навчання (PCO)

Система рейтингових (вагових) балів та критерії оцінювання

Поточний контроль: активність — питання і відповіді на лекційних заняттях, питання і відповіді та участь у ревію на практичних заняттях; МКР; виконання завдань до практичних занять; виконання та представлення робіт комп'ютерного практикуму.

Календарний контроль: провадиться двічі на семестр як моніторинг поточного стану виконання вимог силабусу ("атестація").

Семестровий контроль: залік.

Умови допуску до семестрового контролю: семестровий рейтинг — не менше 40 балів.

Таблиця відповідності рейтингових балів оцінкам за університетською шкалою:

Бали	Оцінка
95 - 100	Відмінно
85 - 94	Дуже добре
75 - 84	Добре
65 - 74	Задовільно
60 - 64	Достатньо
Менше 60	Незадовільно
R < 40	Не допущено

- **Активність**

Активність (частота, змістовність) участі студента у процесі обговорення відповідних тем на заняттях — питання і відповіді на лекційних заняттях, питання і відповіді та участь у ревію на практичних заняттях; виконання завдань до практичних занять; оцінюється, максимум, 20 балами, які може отримати кожен студент за семестр.

- **Роботи комп'ютерного практикуму**

Максимальна кількість балів за виконання кожної роботи комп'ютерного практикуму становить 15 балів.

Критерії оцінювання:

Виконання робіт комп'ютерного практикуму:

- виконана у повному обсязі – 15 балів;
- виконана частково — відповідна частка від максимальної кількості балів.

- Модульна контрольна робота

Максимальна кількість балів за модульну контрольну роботу дорівнює 20 балів.

Якість виконання роботи:

- виконана у повному обсязі з необхідними текстовими поясненнями дій та результатів – максимальна кількість балів;
- виконана частково з поясненнями — відповідна частка від максимальної кількості балів;
- виконана без текстових пояснень дій та результатів – не більше чотирьох балів.

- Творче завдання

Студент може обрати додаткове завдання творчого характеру, результати виконання якого можуть бути опубліковані у наукових виданнях, або повідомлені на студентській науковій конференції з публікацією тез. Максимальна кількість балів за виконання творчого завдання — 30 балів, за умови, що загальна кількість балів не перевищує 100 балів.

Розрахунок шкали (R) рейтингу:

Сума вагових балів контрольних заходів протягом семестру (шкала рейтингу)

складає:

$$R = r_{\text{АКТ}} + r_{\text{КП}} + r_{\text{МКР}} + r_{\text{ТВЗ}} = 20 + 60 + 20 + (\text{до } 30) = 100 \text{ балів.}$$

Робочу програму навчальної дисципліни (силабус) «Логічне програмування»:

Склав доцент кафедри ІІЗЕ, к.т.н., доц. Стативка Юрій Іванович

Ухвалено кафедрою ІІЗЕ (протокол № 28 від 15.05.2023 р)

Погоджено Методичною комісією факультету¹ (протокол № 9 від 26.05.2023 р.)