

**НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ВОДНОГО ГОСПОДАРСТВА ТА  
ПРИРОДОКОРИСТУВАННЯ**

Навчально-науковий інститут автоматики, кібернетики та обчислювальної техніки

**04-04-24S**

**СИЛАБУС**

*навчальної дисципліни*

**SYLLABUS**

Логічне і функціональне програмування		Logical and Functional Programming
Шифр за ОП	<b>ПП 2.3</b>	Code in Degree Programme
Освітній рівень: Магістерський (другий)		Level of Education: Master's (second)
Галузь знань <b>Інформаційні технології</b>	<b>12</b>	Field of Knowledge <b>Information Technology</b>
Спеціальність <b>Комп'ютерна інженерія</b>	<b>123</b>	Field of Study <b>Computer Engineering</b>
Освітня програма: <b>Комп'ютерна інженерія</b>		Degree Programme: <b>Computer Engineering</b>

Силабус навчальної дисципліни *Логічне і функціональне програмування* для здобувачів вищої освіти ступеня «магістр», які навчаються за освітньо-професійною програмою «Комп'ютерна інженерія», спеціальності «Комп'ютерна інженерія», 123. Рівне. НУВГП. 2023. 13 стор.

ОП на сайті університету: <https://ep3.nuwm.edu.ua/22993/>

Розробник силабусу: *Бойчура Михайло Володимирович, к.т.н., старший викладач кафедри обчислювальної техніки*

Силабус схвалений на засіданні кафедри обчислювальної техніки  
Протокол № 13 від "03" червня 2023 року

Завідувач кафедри: *Круліковський Б.Б., к.т.н., доцент.*

Керівник (гарант) ОП: *Круліковський Б.Б., к.т.н., доцент.*

Схвалено науково-методичною радою з якості ННІ АКOT  
Протокол № 8 від "19" червня 2023 року

Голова науково-методичної ради з якості ННІ: *Мартинюк П.М., д.т.н., професор.*

Попередня версія силабусу: відсутня.

ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ	
Логічне і функціональне програмування	
ЗАГАЛЬНА ІНФОРМАЦІЯ	
Ступінь вищої освіти	<i>Магістр</i>
Освітня програма	<i>Комп'ютерна інженерія</i>
Спеціальність	<i>123 Комп'ютерна інженерія</i>
Рік навчання, семестр	<i>1-й рік, 1-й семестр</i>
Кількість кредитів	<i>4,5</i>
Лекції:	<i>22 години</i>
Лабораторні заняття:	<i>24 годин</i>
Самостійна робота:	<i>89 годин</i>
Курсова робота:	<i>Ні</i>
Форма навчання	<i>денна/заочна</i>
Форма підсумкового контролю	<i>Залік</i>
Мова викладання	<i>Державна</i>
ІНФОРМАЦІЯ ПРО РОЗРОБНИКА	
Лектор	 <p><i>Бойчура Михайло Володимирович</i> к.т.н., старший викладач кафедри обчислювальної техніки</p>
Вікіситет	<a href="http://wiki.nuwm.edu.ua/index.php/Бойчура_Михайло_Володимирович">http://wiki.nuwm.edu.ua/index.php/Бойчура Михайло Володимирович</a>
ORCID	<a href="https://orcid.org/0000-0002-9073-4037">https://orcid.org/0000-0002-9073-4037</a>
Канали комунікації	<a href="mailto:m.v.boichura@nuwm.edu.ua">m.v.boichura@nuwm.edu.ua</a> , <i>Телеграм-група студентської групи</i>

ІНФОРМАЦІЯ ПРО НАВЧАЛЬНУ ДИСЦИПЛІНУ
<b>Мета та завдання</b>
<i>Мета: набуття студентами знань, необхідних для формування</i>

теоретичної бази та практичних навичок використання методів і засобів логічного та функціонального програмування для розв'язання складних і неформалізованих задач, що зустрічаються в реальних економічних, організаційних і виробничих системах.

**Завдання:**

- здобути навички розробки застосунків із використанням логіки предикатів;
- навчитись розробляти та розгортати веб-застосунки, що включають роботу з базою даних, авторизацію та автентифікацію із застосуванням саме функціональної парадигми програмування.

**Посилання на розміщення освітнього компонента на навчальній платформі Moodle, на платформі освітніх програм та їхніх освітніх компонентів**

<https://exam.nuwm.edu.ua/course/view.php?id=2742>

**Передумови вивчення  
(місце освітнього компоненту в структурно-логічній схемі)**

Отримані навички можуть використовуватись при написанні кваліфікаційної магістерської роботи (ПП 2.8).

**Компетентності**

СК2. Здатність розробляти алгоритмічне та програмне забезпечення, компоненти комп'ютерних систем та мереж, Інтернет додатків, кіберфізичних систем з використанням сучасних методів і мов програмування, а також засобів і систем автоматизації проектування.

СК3. Здатність проектувати комп'ютерні системи та мережі з урахуванням цілей, обмежень, технічних, економічних та правових аспектів.

СК4. Здатність будувати та досліджувати моделі комп'ютерних систем та мереж.

СК5. Здатність будувати архітектуру та створювати системне і прикладне програмне забезпечення комп'ютерних систем та мереж.

**Програмні результати навчання (ПРН). Результати навчання (РН)\***

РН1. Застосовувати загальні підходи пізнання, методи математики, природничих та інженерних наук до розв'язання складних задач комп'ютерної інженерії.

РН13. Зрозуміло і недвозначно доносити власні знання, висновки та аргументацію з питань інформаційних технологій і дотичних міжгалузевих питань до фахівців і нефахівців, зокрема до осіб, які навчаються.

## СТРУКТУРА ТА ЗМІСТ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

№	Теми (лекції)	Опис лекції	№	Теми лабораторних занять
<b>МОДУЛЬ 1.</b>				
<b>Змістовий модуль 1. Логічне програмування на мові Prolog</b>				
1	Теорія та засоби логічного програмування. 2 год. лекцій PH1, PH13	Імперативні та декларативні мови програмування. Сфери застосування логічних мов програмування. Елементи логіки предикатів. Середовище розробки Visual Prolog v. 5.2.	–	–
2	Програмування на мові Prolog. 2 год. лекцій PH1	Основи синтаксису мови Prolog. Твердження-факт. Твердження-правило. Логічні <i>i, або, не, зворотна імплікація</i> . Coding Conventions. Структура програми на мові Prolog. Розділи Prolog. Стандартні доменні типи.	1	Програма «Чотирикутник». Програма «Конверт». 2 год. лабораторних
3	Детальне вивчення базового синтаксису мови Prolog. 2 год. лекцій PH1	Складені правила. Твердження-відсікання. Алгоритмічні конструкції. Механізм відкату. Точки розгалуження. Уніфікаційні підпрограми Prolog. Змінні у Prolog. Вільна змінна. Змінна зі значенням. Складені домени. Випадки застосування складених доменів. Оголошення домена з альтернативами. Використання доменів з альтернативами.	2	Складені правила. Домени з альтернативами. 2 год. лабораторних
4	Повторювані обчислення. 2 год. лекцій PH1	Способи управління повторюваними обчисленнями. Механізм відкату. Способи управління відкатом. Взаємодія відкату після невдачі та відкату із відсіканням. Реалізація меню в Prolog. Відлагодження програми на Prolog. Рекурсія. Випадки, які приводять до використання рекурсії. Повторення на прямому ході рекурсії. Повторення	3-4	Повторювані обчислення. 4 год. лабораторних

		на зворотному ході рекурсії. Приклади обчислення факторіалу, суми з умовами, суми ряду.		
5	Списки. 2 год. лекцій PH1	Синтаксис роботи зі списками. Оголошення списку. Цільові запити до списків. Розділення списку та голову та хвіст. Основні алгоритми роботи зі списками. Наповнення, вивід, пошук, вставка та видалення елементів.	5	Елементарні дії над елементами списку. 2 год. лабораторних
6	Рядки. 2 год. лекцій PH1	Вступ. Стандартні твердження для роботи з рядками. Твердження str_len, concat, frontchar, frontstr, str-int, str-real, str-char, upper-lower, isname, fronttoken. Основні алгоритми для роботи з рядками. Пошук, вставка, видалення, конвертація символів/рядків.	6	Сортування елементів списку. 2 год. лабораторних
<b>МОДУЛЬ 2.</b>				
<b>Змістовий модуль 2. Функціональне програмування на мові Go</b>				
7	Вступ до мови Go. 2 год. лекцій PH13	Вступ. Структура офіційного сайту мови Go. Призначення мови Go. Документація мови Go. Перша програма на мові Go. Структура програми. Пакет pkg. Введення та виведення даних.	–	–
8-9	Розробка та публікація веб-сайту. 4 год. лекцій PH13	Пакети net/http, html/template. Структура веб-додатку. Запуск локального сервера. Система маршрутизації. Передача даних на веб-сторінку. Bootstrap. Файлова структура веб-додатку. Доступ до статичних файлів CSS та JavaScript. Створення шаблонів. Використання шаблонів. Шаблонізатори. http.Handler. Передача масивів на html-сторінку. Умовні конструкції в шаблонізаторах. Оператори в шаблонізаторах.	7-8	Розробка та публікація веб-сайту з використанням фреймворку Bootstrap та мови Go. 4 год. лабораторних

10	Робота з базами даних. 2 год. лекцій PH1, PH13	Створення бази даних. Підключення до бази даних. Додавання, зчитування, редагування та видалення даних з бази даних. Обмін даними між базою даних та html-сторінкою. Об'єкт типу *http.Request. Відправлення форм. Методи POST та GET. Роутер Gin.	9-10	Інтеграції баз даних у веб-додаток. 4 год. лабораторних
11	Авторизація та автентифікація. 2 год. лекцій PH1, PH13	Маршрутизація gorilla/mux. Рядок запиту. REST API. Структура JWT. Header. Payload. Signature. Авторизація. Автентифікація. Ролі.	11-12	Реалізація авторизації та автентифікації користувачів. 4 год. лабораторних

### Форми, методи та технології навчання

Форми навчання	<ul style="list-style-type: none"> <li>• очна (денна) з, можливо, елементами дистанційного навчання;</li> <li>• заочна.</li> </ul>
Форми навчального процесу	<ul style="list-style-type: none"> <li>• навчальні заняття (лекції, лабораторні заняття, консультації);</li> <li>• самостійна робота здобувачів;</li> <li>• робота в наукових бібліотеках та мережі Інтернет;</li> <li>• контрольні заходи (поточна складова оцінювання, модульні контролю, підсумковий контроль).</li> </ul>
Методи та технології навчання	<ul style="list-style-type: none"> <li>• робота в малих групах (команді) та індивідуальна робота;</li> <li>• проектна технологія;</li> <li>• аналіз конкретних ситуацій (case study): ситуація-оцінка;</li> <li>• контекстне навчання;</li> <li>• проблемне навчання.</li> </ul>
Процес навчання включає, зокрема, наступне	<ul style="list-style-type: none"> <li>• написання комп'ютерних програм;</li> <li>• відлагодження програм;</li> <li>• Code Review;</li> <li>• слідування рекомендаціям Coding Conventions.</li> </ul>
Засоби навчання	<ul style="list-style-type: none"> <li>• відео-запис лекції;</li> </ul>

- презентація;
- підручник;
- конспект лекцій;
- різні туторіали.

### **Інструменти, обладнання, програмне забезпечення**

- середовище розробки: *Visual Prolog 5.2*;
- редактор вихідного коду: *Microsoft Visual Studio Code* (з розширенням *Go*);
- *Git* термінал: *GitHub Desktop*;
- Хостинг: <https://onrender.com/>.

### **Порядок оцінювання програмних результатів навчання/ результатів навчання**

Студент може отримати сумарно не більше, ніж 100 балів, за наступні складові:

1. модульні контролі: 40 балів;
2. поточний контроль: 50-60 балів;
3. додаткові бали: 0-10 балів.

Розподіл балів:

1. за модульні контрольні роботи:

- модульний контроль №1 (20 балів):  
Рівень 1 – 19 запитань по 0.5 балів за кожне.  
Рівень 2 – 6 запитань по 0.9 балів за кожне.  
Рівень 3 – 3 запитання по 1.7 балів за кожне.
- модульний контроль №2 (20 балів):  
Рівень 1 – 19 запитань по 0.5 балів за кожне.  
Рівень 2 – 6 запитань по 0.9 балів за кожне.  
Рівень 3 – 3 запитання по 1.7 балів за кожне.

2. за лабораторні роботи (50-60 балів):

Передбачено по 5 балів за кожну лабораторну роботу; у випадку правильного виконання лабораторної роботи оцінка лінійно залежить від відсотка розуміння коду. Як альтернатива, студенти можуть виконувати завдання на інших мовах/середовищах логічного/функціонального програмування за умови попереднього узгодження деталей з викладачем.

3. додаткові бали за вагому громадянську та студентську активність (0-10 балів):

Виставляється до 10 балів за волонтерство, олімпіади, спартакіади, конкурси, конференції, написання статей, активну студентську діяльність, конкретні пропозиції з удосконалення змісту навчальної дисципліни тощо.

### **Поєднання навчання та досліджень**

Мову *Go* називають мовою стартаперів, оскільки вона дозволяє



отримувати швидкий та якісний результат при порівняно невеликих зусиллях. Таким чином, студенти можуть досить швидко робити свої дослідження загальнодоступними, викладаючи їх на власному сайті в мережі Інтернет.

У певних випадках можна зекономити час на формалізацію задачі, застосувавши мову Prolog.

За виконання індивідуальних завдань дослідницького характеру студенти мають можливість додатково отримати бали, а також можуть бути долучені до написання та публікації наукових статей з тематики дисципліни.

### Рекомендована література (основна, допоміжна)

#### Основна література

1. Заяць В. М., Заяць М. М. Логічне і функціональне програмування. Системний підхід: підручник (2-ге видання, випр. та доповн.). Рівне: Національний університет водного господарства та природокористування. 2018. 422 с.

2. Дейнега Л. Ю., Камінська Ж. К., Левада І. В., Сердюк С. М. Практичне програмування мовою Visual Prolog : навч. посіб. Запоріжжя : ЗНТУ, 2016. 236 с.

3. Темнікова О. Л. Математична логіка та теорія алгоритмів: конспект лекцій : навч. посіб. Київ : КПІ ім. Ігоря Сікорського, 2021. 177 с.

4. Безсмертний І. А. Інтелектуальні системи. [електронний ресурс]. URL: [https://stud.com.ua/139940/informatika/intelektualni\\_sistemi](https://stud.com.ua/139940/informatika/intelektualni_sistemi) (Last access: 14.01.2023).

5. Buscaroli R., Chesani F., Giuliani G., Loreti D., Mello P. A Prolog application for reasoning on maths puzzles with diagrams. *Journal of Experimental & Theoretical Artificial Intelligence*. 2022. DOI: 10.1080/0952813X.2022.2062456.

6. Зуєв А.О., Сальніков Д.В., Васильченков О.Г. Методичні вказівки для виконання самостійних робіт «Основи програмування на мові Golang». Харків: НТУ «ХПІ», 2022. 35 с.

7. Wampler D. *Programming Scala*. Third Edition. Sebastopol: O'Reilly Media, Inc. 2021. 520 p.

#### Допоміжна література

1. A Tour of Go. URL: <https://go.dev/tour/welcome/1> (Last accessed: 02.06.2023).

2. Documentation - The Go Programming Language. URL: <https://go.dev/doc/> (Last accessed: 02.06.2023).

3. Go by Example. URL: <https://gobyexample.com/> (Last accessed: 02.06.2023).

4. Standard library - Go Packages. URL: <https://pkg.go.dev/std> (Last accessed: 02.06.2023).

5. Introduction | Tour of Scala | Scala Documentation. URL: <https://docs.scala-lang.org/tour/tour-of-scala.html> (Last accessed: 02.06.2023).

6. A Scala Tutorial for Java Programmers | Scala Documentation. URL: <https://docs.scala-lang.org/tutorials/scala-for-java-programmers.html> (Last accessed: 02.06.2023).

7. Introduction | Scala 3 — Book | Scala Documentation. URL: <https://docs.scala-lang.org/scala3/book/introduction.html> (Last accessed: 02.06.2023).

8. What is F# | Microsoft Learn. URL: <https://learn.microsoft.com/en-us/dotnet/fsharp/what-is-fsharp> (Last accessed: 02.06.2023).

### Інформаційні ресурси в Інтернеті

1. <https://www.youtube.com/watch?v=8XUutFBbUrg&list=PLEvH6T-1oh772BDqVCvq0wKkJxwt9iXf7> (The Discipline of Prolog).
2. <https://www.youtube.com/watch?v=Uska9DgJEEoo&list=PLEvH6T-1oh76PGIiCiJ8zEzbQ2RgLFsyL> (Prolog Style and Technique).
3. [https://www.youtube.com/watch?v=jOAR\\_pAPP90&list=PLBPdtL8DZBZL-HZfbLycJo1Gomw46mjDn](https://www.youtube.com/watch?v=jOAR_pAPP90&list=PLBPdtL8DZBZL-HZfbLycJo1Gomw46mjDn) (Prolog :: Logical Programming Course).
4. <https://www.youtube.com/watch?v=Uzv9aj2aPj0&list=PLKdU0fuY4OFELM-IEmZYLe3eHEUISGru5> (Prolog Bangla Tutorial).
5. <https://www.youtube.com/playlist?list=PLdo4fOcmZ0oV2uhVlItfXxuRXsv-gXD5> (.NET Conf: Focus on F#).
6. <https://www.youtube.com/playlist?list=PLpsileTZltjWrPQjTzJ8Fj2ah7JGBCss5> (Track Introductions).
7. <https://www.youtube.com/watch?v=un6ZyFkqFKo> (Go Programming – Golang Course with Bonus Projects).

### ПОЛІТИКИ ВИКЛАДАННЯ ТА НАВЧАННЯ

#### Перелік соціальних, «м'яких» навичок (soft skills)

Вміння комунікувати	<ul style="list-style-type: none"><li>• здатність спілкуватися державною мовою як усно, так і письмово;</li><li>• вміння спілкуватись та писати із використанням англomовної професійної термінології;</li><li>• навички усного спілкування;</li><li>• навички письмового спілкування;</li><li>• вміння писати зрозумілий код.</li></ul>
Вміння сумісно працювати	<ul style="list-style-type: none"><li>• вміння управляти часом;</li><li>• навички управління проектами;</li></ul>

	<ul style="list-style-type: none"> <li>• здатність планувати свій час у плані співставлення вимог, власних знань, здібностей і дедлайнів;</li> <li>• здатність працювати в команді;</li> <li>• навички міжособистісних відношень;</li> <li>• вміння надавати рекомендації іншим у коректній формі.</li> </ul>
Здатність до аналізу та синтезу	<ul style="list-style-type: none"> <li>• здатність критично мислити;</li> <li>• знаходити вихід з складних ситуацій;</li> <li>• здатність до навчання;</li> <li>• комплексне рішення проблем;</li> <li>• критичне мислення.</li> </ul>
Здатність застосовувати знання у практичних ситуаціях	

### **Дедлайни та перескладання**

Дедлайн здачі лабораторних робіт – до кінця сесії. Здача лабораторних робіт відбувається на парі або під час консультації, дата та час якої гнучко узгоджується між студентом та викладачем.

На задачу кожного з модульних контролів студенту надається одна спроба. Перший модуль здається на будь-якій лекції у квітні, а другий – на передостанній чи останній лекції. Перездача окремого модульного контролю передбачена лише за виключних обставин. При бажанні покращити оцінку за модульну складову оцінювання студент під час сесії звертається до викладача з проханням здати підсумковий контроль (40 балів). При цьому, попередні бали за модульні контролі анулюються.

У разі, якщо здобувач не набрав 60 балів після закінчення сесії, його відправляють на комісію з ліквідації академічної заборгованості. Якщо і тоді здобувач не набирає необхідної кількості балів, то передбачається повторний курс.

### **Неформальна та інформальна освіта**

Студенти мають право на часткове або повне перезарахування предмету за умови написання ними відповідної заяви та надання документів, які підтверджують ті результати навчання, які здобувач отримав (див. положення <https://ep3.nuwm.edu.ua/18660/>). Зокрема студенти можуть самостійно проходити онлайн-курси на таких навчальних платформах, як Prometheus, Coursera, edEx, edEra, FutureLearn та інших, для наступного перезарахування результатів навчання. Проте доцільно попередньо узгодити з викладачем відповідність обраного онлайн-курсу суті навчальної дисципліни. Деякий перелік підходящих курсів наведено нижче:

- Coursera – Getting Started with Go (Початок роботи з Go);

- Coursera – *Functional Programming in Scala* (Функціональне програмування в Scala);
- Coursera – *Kotlin for Java Developers* (Kotlin для розробників Java);
- *Exercism – Prolog*;
- *Swayam – Artificial Intelligence: Knowledge Representation And Reasoning* (Штучний інтелект: представлення знань і міркування);
- *Pluralsight – Code School: On Track with Golang 1* (Школа коду: на шляху до Golang 1);
- *Pluralsight – F# 6 Fundamentals* (Основи F# 6).

Пошук курсів у зручній формі доступний тут: <https://www.classcentral.com/>.

Окрім того, якщо з'являються обставини для здобуття неформальної чи інформальної освіти від викладачів-практиків, то пропонуються ці можливості для студентів; рекомендуються відео-уроки практикуючих програмістів з Youtube тощо.

### **Правила академічної доброчесності**

Задля запобігання академічної недоброчесності вимагається наступне:

- кожен студент у групі виконує завдання згідно запропонованого йому варіанту або пропонує свою тему, яку обов'язково узгоджує з викладачем;
- студент отримує хоч якусь оцінку лише за умови розуміння коду програми;
- студентам забороняється: плагіатити, самоплагіатити, фабрикувати, фальсифікувати, списувати, обманювати та будь-яким чином впливати на викладача, включаючи спроби хабарництва.

Залежно від виду та ступеня порушення викладач може накладати наступні санкції:

- усне або письмове зауваження від викладача;
- попередження про можливість притягнення до академічної відповідальності;
- зниження чи анулювання результатів оцінювання навчального завдання здобувача вищої освіти;
- повторне виконання навчального завдання;
- виконання іншого навчального завдання;
- призначення додаткового навчання з питань академічної доброчесності;
- призначення додаткових контрольних заходів (додаткові індивідуальні навчальні завдання, тести тощо);
- подання клопотання на ім'я ректора з метою порушення формальної процедури розгляду питання про притягнення

студента до відповідальності.

За списування під час проведення модульного чи підсумкового контролю студент позбавляється подальшого права здавати матеріал і у нього виникає академічна заборгованість.

Документи стосовно академічної доброчесності (про плагіат, порядок здачі курсових робіт, кодекс честі студентів, документи Національного агентства стосовно доброчесності) наведені на сторінці «Якість освіти» офіційного сайту НУВГП – <http://nuwm.edu.ua/sp/akademichna-dobrochesnistj>.

### **Вимоги до відвідування**

Санкції за пропуски пар не передбачені. Студент має право самостійно вивчити необхідний для здачі модульних контролів та лабораторних робіт матеріал, який в повному обсязі дублюється викладачем одночасно на платформі Moodle та/або у групі з даного предмету в месенджері Telegram. Також викладач розміщує відеозаписи пар на Youtube. У разі необхідності проведення консультації – викладач йде назустріч.

Відвідування пари допускається із використанням власного ноутбука. Студенти не повинні порушувати дисципліну на парі.

Для студентів, які знаходяться на індивідуальному плані навчання, надаються індивідуальні завдання.

Автор  
Старший викладач

Михайло БОЙЧУРА

Затверджено

Проректор з науково-педагогічної та  
навчальної роботи

Валерій СОРОКА



документ підписаний КЕП  
Номер документа СИЛ №524 від [sDateTime\_SignWriteAgree\_Last]  
Підписувач Сорока Валерій Степанович  
Підписувач (дані КЕП): [oSignECP.sSigner\_Sert]  
Сертифікат 58E2D9E7F900307B0400000807E2D0054327D00