

**КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ ТОРГОВЕЛЬНО-ЕКОНОМІЧНИЙ  
УНІВЕРСИТЕТ**  
**СИСТЕМА УПРАВЛІННЯ ЯКІСТЮ**  
Система забезпечення якості освітньої діяльності та якості вищої освіти  
*сертифікована на відповідність ДСТУ ISO 9001:2015 / ISO 9001:2015*  
Кафедра інженерії програмного забезпечення та кібербезпеки

## **СИЛАБУС**

### **ФУНКЦІОНАЛЬНЕ ТА ЛОГІЧНЕ ПРОГРАМУВАННЯ /**

### **FUNCTIONAL AND LOGICAL PROGRAMMING**

### **SYLLABUS**

<b>освітній ступінь</b>	<b>магістр</b>	<b>/</b>	<b>master</b>
<b>галузь знань</b>	<b>12 Інформаційні технології</b>	<b>/</b>	<b>Information Technologies</b>
<b>спеціальність</b>	<b>121 Інженерія програмного забезпечення</b>	<b>/</b>	<b>Software Engineering</b>

**Київ 2021**

**Розповсюдження і тиражування без офіційного дозволу КНТЕУ  
заборонено**

Автор Т.В. Савченко, канд. техн. наук, доцент

Силабус розглянуто та схвалено на засіданні кафедри інженерії програмного забезпечення та кібербезпеки «27» серпня 2021 р., протокол № 1.

**ФУНКЦІОНАЛЬНЕ ТА ЛОГІЧНЕ ПРОГРАМУВАННЯ /  
FUNCTIONAL AND LOGICAL PROGRAMMING**

**СИЛАБУС / SYLLABUS**

<b>освітній ступінь</b>	<b>магістр</b>	<b>/</b>	<b>master</b>
<b>галузь знань</b>	<b>12 Інформаційні технології</b>	<b>/</b>	<b>Information Technologies</b>
<b>спеціальність</b>	<b>121 Інженерія програмного забезпечення</b>	<b>/</b>	<b>Software Engineering</b>

Автор САВЧЕНКО Тетяна Віталіївна

## АНОТАЦІЯ КУРСУ

### 1. Викладач:

#### **1.1. Лектор:** Савченко Тетяна Віталіївна

- *вчений ступінь, вчене звання та посада:* кандидат технічних наук, доцент, доцент кафедри інженерії програмного забезпечення та кібербезпеки;
- *педагогічний стаж* – 23 роки;
- *e-mail:* [savchenko\\_tv@knu.edu.ua](mailto:savchenko_tv@knu.edu.ua)
- *наукові інтереси:* інформаційні технології та системи, кібербезпека, програмування, оптимізація;
- *стажування та підвищення кваліфікації:*
  - ✓ «Build Your Own Chatbot» (CB0103EN, provided by Cognitive Class), a course on [bdu.intela-edu.com](https://bdu.intela-edu.com). Powered by IBM Developer Skills Network. Issued by Intela-EDU. Issued on: April 23, 2019. Authenticity of this certificate can be validated by going to: <https://courses.bdu.intela-edu.com/certificates/e9b3aef3b72f46c4a7a7d1d43e52bda3>
  - ✓ IT Ukraine Association Teacher's Internship program held by EPAM Systems, period June-August 2020 (108 hours).
  - ✓ CISCO: Introduction Cybersecurity (09.06.2020); Cybersecurity Essentials (23.09.2020); CCNA Cybersecurity Operations (03.10.2020); CCNA Security (28.10.2020); CCNAv7: Introduction to Networks (15.02.2021).
  - ✓ Курси підвищення кваліфікації у Державному університеті телекомунікацій за темою: «Системи технічного захисту інформації» (120 годин), 23.11-04.12.2020 р.

#### **1.1. Дисципліна:** «Функціональне та логічне програмування»:

- рік навчання: 1 рік магістратури;
- семестр навчання: 1;
- кількість кредитів: 6;
- кількість годин за семестр:
  - лекційних: 28 год.;
  - лабораторних: 28 год.;
  - на самостійне опрацювання: 124 год.
- кількість аудиторних годин на тиждень:
  - лекційних: 2 год.;
  - лабораторних: 2 год.

### 2. Час та місце проведення:

- аудиторні заняття – відповідно до розкладу КНТЕУ, з урахуванням специфіки дисципліни передбачено аудиторіях: 510, 514.

- позааудиторна робота – самостійна робота студента, результат виконання якої висвітлено засобами Office 365.

### 3. Пререквізити та постреквізити навчальної дисципліни:

- **пререквізити:** «Об'єктно-орієнтоване програмування», «Web-програмування та Web-дизайн», «Алгоритми та структури даних»;
- **постреквізити:** набуття навичок застосування функціонального та логічного програмування (зокрема, мов програмування високого рівня LISP та PROLOG) в майбутній професійній діяльності.

### 4. Результати вивчення дисципліни:

Дисципліна «Функціональне та логічне програмування», як вибіркова компонента освітньої програми, забезпечує оволодіння студентами загальними та фаховими компетентностями і досягнення ними програмних результатів навчання за відповідною освітньо-професійною програмою:

#### *Інженерія програмного забезпечення (ОС магістр)*

Номер в освітній програмі	Зміст компетентності	Номер теми, що розкриває зміст компетентності
<i>Загальні компетентності за освітньою програмою</i>		
ЗК01	Здатність до абстрактного мислення, аналізу та синтезу.	1 – 14
ЗК03	Здатність проводити дослідження на відповідному рівні.	1 – 14
ЗК05	Здатність генерувати нові ідеї (креативність).	1 – 14
<i>Фахові компетентності за освітньою програмою</i>		
СК01	Здатність аналізувати предметні області, формувати, класифікувати вимоги до програмного забезпечення.	1 – 14
СК07	Здатність критично осмислювати проблеми у галузі інформаційних технологій та на межі галузей знань, інтегрувати відповідні знання та розв'язувати складні задачі у широких або мультидисциплінарних контекстах.	1 – 14
СК08	Здатність розробляти і координувати процеси, етапи та ітерації життєвого циклу програмного забезпечення на основі застосування сучасних моделей, методів та технологій розроблення програмного забезпечення.	1 – 14
СК09	Здатність забезпечувати якість програмного забезпечення.	1 – 14
<i>Програмні результати навчання за освітньою програмою</i>		
РН01	Знати і застосовувати сучасні професійні стандарти і інші нормативно-правові документи з інженерії програмного забезпечення.	1 – 14
РН02	Оцінювати і вибирати ефективні методи і моделі розроблення, впровадження, супроводу програмного забезпечення та управління відповідними процесами на всіх етапах життєвого циклу.	1 – 14
РН04	Виявляти інформаційні потреби і класифікувати дані для проектування програмного забезпечення.	1 – 14
РН09	Обґрунтовано вибирати парадигми і мови програмування для розроблення програмного забезпечення; застосовувати на практиці сучасні засоби розроблення програмного	1 – 14

	забезпечення.	
<i>PH10</i>	Модифікувати існуючі та розробляти нові алгоритмічні рішення детального проектування програмного забезпечення.	1 – 14
<i>PH11</i>	Забезпечувати якість на всіх стадіях життєвого циклу програмного забезпечення, у тому числі з використанням релевантних моделей та методів оцінювання, а також засобів автоматизованого тестування і верифікації програмного забезпечення.	1 – 14
<i>PH13</i>	Конфігурувати програмне забезпечення, керувати його змінами та розробленням програмної документації на всіх етапах життєвого циклу.	1 – 14
<i>PH15</i>	Здійснювати реінжиніринг програмного забезпечення відповідно до вимог замовника.	1 – 14

## 5. Характеристика дисципліни:

**5.1. Призначення навчальної дисципліни:** вивчення дисципліни дозволить студентам оволодіти методами функціонального та логічного програмування, теоретичними знаннями щодо принципів функціонального підходу при розробці програм; основними підходами і засобами логічного програмування з використанням різноманітних структур даних у програмуванні та їх застосуванні при побудові ефективних алгоритмів; вмінням користуватись сучасним програмним забезпеченням.

**5.2. Мета вивчення дисципліни:** полягає у набутті студентами знань, необхідних для формування теоретичної бази знань та практичних навичок використання методів та засобів функціонального та логічного програмування для розв'язання складних і неформалізованих задач, що зустрічаються в реальних економічних, організаційних і виробничих системах.

**5.3. Задачі вивчення дисципліни:** є теоретична та практична підготовка майбутніх фахівців з таких питань:

- набуття знань з теоретичних основ в галузі побудови, розробки та експлуатації програмних продуктів, що використовують засоби логічного та функціонального програмування;
- ознайомлення із сучасним станом та перспективами розвитку функціонального та логічного програмування;
- оволодіння практичними навичками створення програмних продуктів засобами логічного та функціонального програмування, застосування відповідних методів, моделей і алгоритмів розв'язання інтелектуальних задач.

**5.4. Зміст навчальної дисципліни:** відповідає навчальній та робочій програмі, що визначено науковими досягненнями як вітчизняних, так і закордонних вчених, а також запитам стейкхолдерів.

**5.5. План вивчення дисципліни:**

Таблиця

Навчальна діяльність*	Роб. час студ., год.	Оцінювання, бали
<p align="center"><b>РОЗДІЛ 1. ФУНКЦІОНАЛЬНЕ ПРОГРАМУВАННЯ</b></p> <p><b>Тема 1. Загальне уявлення про функціональне програмування та його застосування</b> <i>План лекції</i></p> <ol style="list-style-type: none"> <li>Домінуючі парадигми програмування.</li> <li>Функціональне та логічне програмування.</li> <li>Концепція функціонального програмування.</li> <li>LISP – мова програмування високого рівня.</li> <li>Особливості функціонального програмування.</li> </ol> <p align="center"><b>Список рекомендованих джерел:</b></p> <p>Основний: 1 Додатковий: 4, 5 Інтернет-ресурси: 11</p> <p><b>Самостійна робота:</b> ознайомлення з основними парадигмами програмування: імперативне, директивне, структурне, непроцедурне, об'єктно-орієнтоване, декларативне програмування; підготовка до лабораторного заняття № 1.</p> <p align="center"><b>Лабораторне заняття № 1</b> <b>Основні конструкції мови LISP</b></p> <p><b>Мета:</b> ознайомлення та засвоєння основних команд та конструкцій мови LISP; набуття навичок практичного застосування знань.</p> <p><b>Завдання:</b> скласти програму обчислення значення функції згідно з варіантом; засвоїти основні команди та конструкції мови LISP.</p>	<p align="center">2</p> <p align="center">8</p> <p align="center">2</p>	<p align="center">3</p> <p align="center">4</p>
<p align="center"><b>Тема 2. Елементарний LISP</b> <i>План лекції</i></p> <ol style="list-style-type: none"> <li>Основні визначення.</li> <li>Списки як засіб подання знань.</li> <li>Вбудовані функції LISP.</li> <li>Предикативна форма запису функцій.</li> <li>Базові функції обробки списків.</li> <li>Елементарні функції над списками та S-виразами.</li> </ol> <p align="center"><b>Список рекомендованих джерел:</b></p> <p>Основний: 1 Додатковий: 4, 5 Інтернет-ресурси: 11</p> <p><b>Самостійна робота:</b> ознайомлення з основними функціями обробки списків у мові LISP: функції CAR і CDR, функція-конструктор CONS, предикати ATOM, EQ, EQUAL; підготовка до лабораторного заняття № 2.</p>	<p align="center">2</p> <p align="center">8</p>	<p align="center">3</p>

Навчальна діяльність*	Роб. час студ., год.	Оцінювання, бали
<p align="center"><b>Лабораторне заняття № 2</b> <b>Застосування функцій на мові LISP</b></p> <p><b>Мета:</b> вивчення базових функцій організації та обробки списків, а також способів опису та виклику нерекурсивних функцій у мові LISP; отримання навичок роботи з інтерпретатором LISP.</p> <p><b>Завдання:</b> вивчити роботу примітивних базових функцій; розробити текст програми згідно варіанту, записати в редакторі LISP та перевірити працездатність програми.</p>	2	4
<p align="center"><b>Тема 3. Конструювання списків</b> <i>План лекції</i></p> <ol style="list-style-type: none"> <li>1. Конструювання списків у LISP.</li> <li>2. Функції вищого рівня.</li> <li>3. Функції виділення елементів списку.</li> <li>4. Неіменовані функції LISP.</li> <li>5. Застосування <i>lambda</i>-виразів та <i>nlambda</i>-виразів.</li> <li>6. Іменовані функції LISP.</li> </ol> <p align="center"><b>Список рекомендованих джерел:</b></p> <p>Основний: 1 Додатковий: 4, 5 Інтернет-ресурси: 11</p> <p><b>Самостійна робота:</b> ознайомлення з функціями конструювання списків у LISP (CONS, LIST, APPEND), виділення елементів списку (LENGTH, REVERSE, LAST, NTH), функціями DEFUN та LABEL; підготовка до лабораторного заняття № 3.</p>	2	
<p align="center"><b>Лабораторне заняття № 3</b></p> <p><b>Основні конструкції мови LISP. Структури розгалуження</b></p> <p><b>Мета:</b> вивчення основних конструкцій мови LISP, функцій розгалуження; отримання навичок використання конструкції COND.</p> <p><b>Завдання:</b> вивчити роботу функцій розгалуження; розробити текст програми з використанням конструкції COND згідно варіанту, записати в редакторі LISP та перевірити працездатність програми.</p>	8	3
	2	4

Навчальна діяльність*	Роб. час студ., год.	Оцінювання, бали
<p align="center"><b>Тема 4. Числові функції. Керуючі структури</b> <i>План лекції</i></p> <ol style="list-style-type: none"> <li>Числові функції в LISP.</li> <li>Арифметичні, логічні, ірраціональні та трансцендентні функції.</li> <li>Тригонометричні функції.</li> <li>Керуючі структури.</li> <li>Структури розгалуження.</li> <li>Циклічні обчислення в LISP.</li> </ol> <p align="center"><b>Список рекомендованих джерел:</b></p> <p>Основний: 1 Додатковий: 4, 5</p> <p><b>Самостійна робота:</b> ознайомлення з функціями присвоєння SET, SETQ, LET; функціями розгалуження COND, IF, WHEN і UNLESS; функціями циклічних обчислень DO, DOTIMES, LOOP; підготовка до лабораторного заняття № 4.</p> <p align="center"><b>Лабораторне заняття № 4</b> <b>Числові функції в LISP. Циклічні обчислення</b></p> <p><b>Мета:</b> вивчення основних конструкцій мови LISP, функцій циклічних обчислень; отримання навичок практичного використання циклічних конструкцій.</p> <p><b>Завдання:</b> вивчити роботу циклічних функцій; розробити текст програми з використанням конструкції з циклічними обчисленнями згідно варіанту, записати в редакторі LISP та перевірити працездатність програми.</p>	<p align="center">2</p> <p align="center">8</p> <p align="center">2</p>	<p align="center">3</p> <p align="center">5</p>
<p align="center"><b>Тема 5. Поняття рекурсії. Функціонал</b> <i>План лекції</i></p> <ol style="list-style-type: none"> <li>Прості рекурсії та їх застосування.</li> <li>Правила побудови рекурсивних функцій.</li> <li>Рекурсії вищих порядків.</li> <li>Поняття функціонала.</li> <li>Алгоритми пошуку на LISP.</li> </ol> <p align="center"><b>Список рекомендованих джерел:</b></p> <p>Основний: 1 Додатковий: 4, 5 Інтернет-ресурси: 11</p> <p><b>Самостійна робота:</b> ознайомлення з призначенням рекурсивних функцій, аналіз та порівняння різних їх видів; застосування APPLY, FUNCALL, EVAL; підготовка до лабораторного заняття № 5.</p>	<p align="center">2</p> <p align="center">10</p>	<p align="center">3</p>



Навчальна діяльність*	Роб. час студ., год.	Оцінювання, бали
<p align="center"><b>Лабораторне заняття № 5</b></p> <p><b>Застосування простих рекурсивних функцій в мові LISP</b>  <b>Мета:</b> ознайомлення та засвоєння основних правил написання рекурсивних функцій у функціональній мові LISP.  <b>Завдання:</b> виконати завдання згідно з варіантом, використовуючи правила написання рекурсивних функцій; навчитися формулювати умову завершення рекурсії, описувати формування результату функції та нових значень аргументів для рекурсивного виклику.</p>	2	4
<p align="center"><b>Тема 6. Застосування діалектів мови LISP</b></p> <p align="center"><i>План лекції</i></p> <ol style="list-style-type: none"> <li>1. Реалізації стандарту Common LISP.</li> <li>2. Текстовий редактор Emacs.</li> <li>3. Розробка програм в середовищі Scheme.</li> <li>4. Використання Visual LISP в системі AutoCAD.</li> <li>5. Призначення та можливості AutoLISP.</li> </ol> <p align="center"><b>Список рекомендованих джерел:</b></p> <p>Основний: 1  Додатковий: 4, 5  Інтернет-ресурси: 11, 12</p>	2	
<p><b>Самостійна робота:</b> аналіз та порівняння практичного застосування діалектів мови LISP; наведення прикладів реалізації Common LISP, Emacs, Scheme, Visual LISP; підготовка до лабораторного заняття № 6.</p>	10	2
<p align="center"><b>Лабораторне заняття № 6</b></p> <p align="center"><b>Знайомство з середовищем Visual LISP</b></p> <p><b>Мета:</b> ознайомлення з особливостями середовища Visual LISP; оволодіння навичками програмування графічних примітивів у системі AutoCAD.  <b>Завдання:</b> вивчити основні функції та структуру програми на мові AutoLISP; розглянути застосування редактора Visual LISP в системі AutoCAD на простих прикладах, наданих викладачем.</p>	2	5

Навчальна діяльність*	Роб. час студ., год.	Оцінювання, бали
<p><b>Тема 7. Вирішення задач, заснованих на знаннях</b> <i>План лекції</i></p> <ol style="list-style-type: none"> <li>1. Области застосування експертних систем.</li> <li>2. Основи методології розробки експертних систем.</li> <li>3. Експертні системи реального часу.</li> <li>4. CLIPS – програмне середовище для розробки ЕС.</li> <li>5. Основні функції та базові команди CLIPS.</li> <li>6. Використання CLIPS в експертних системах.</li> </ol> <p><b>Список рекомендованих джерел:</b> Основний: 1 Додатковий: 4, 5, 6</p> <p><b>Самостійна робота:</b> ознайомлення з програмним середовищем CLIPS та його практичним застосуванням; використання Бекусо-Наурова (БНФ)-нотації для визначення конструкцій мови CLIPS; підготовка до лабораторного заняття № 7.</p> <p><b>Лабораторне заняття № 7</b> <b>Створення програмного коду в редакторі Visual LISP</b> <b>Мета:</b> набуття досвіду створення програми в редакторі Visual LISP; оволодіння навичками програмування графічних примітивів у системі AutoCAD. <b>Завдання:</b> написати програмний код на мові AutoLISP згідно варіанту, виданого викладачем, застосовуючи знання та навички програмування графічних примітивів у редакторі Visual LISP в системі AutoCAD.</p>	<p>2</p> <p>10</p> <p>2</p>	<p></p> <p>3</p> <p>4</p>
<p><b>РОЗДІЛ 2. ЛОГІЧНЕ ПРОГРАМУВАННЯ</b> <b>Тема 8. Математичні основи логічного програмування</b> <i>План лекції</i></p> <ol style="list-style-type: none"> <li>1. Концепція логічного програмування.</li> <li>2. Основні конструкції логічного програмування.</li> <li>3. Логіка висловлювань.</li> <li>4. Розв'язання логічних задач засобами логіки висловлювань.</li> <li>5. Формалізація процесу логічного виведення.</li> </ol> <p><b>Список рекомендованих джерел:</b> Основний: 2 Додатковий: 7, 8</p> <p><b>Самостійна робота:</b> ознайомлення з практичним застосуванням конструкцій логічного програмування та логіки висловлювань; вивчення формул логіки висловлювань та скороченого методу перевірки аргументів; підготовка до лабораторного заняття № 8.</p>	<p>2</p> <p>8</p>	<p></p> <p>2</p>



Навчальна діяльність*	Роб. час студ., год.	Оцінювання, бали
<p align="center"><b>Тема 10. Особливості мови Visual Prolog</b> <i>План лекції</i></p> <ol style="list-style-type: none"> <li>1. Основні елементи мови Visual Prolog.</li> <li>2. Стандартні типи даних.</li> <li>3. Факти та правила у Visual Prolog.</li> <li>4. Поняття аргументів та предикатів.</li> <li>5. Призначення запитів у Prolog.</li> <li>6. Дерева пошуку рішень.</li> </ol> <p align="center"><b>Список рекомендованих джерел:</b> Основний: 2, 3 Додатковий: 8, 9 Інтернет-ресурси: 13</p> <p><b>Самостійна робота:</b> ознайомлення з робочим середовищем Visual Prolog, поняттям термів, стандартних типів даних (symbol, string, char, integer, real); підготовка до лабораторного заняття № 10.</p> <p align="center"><b>Лабораторне заняття № 10</b> <b>Управління пошуком рішень</b></p> <p><b>Мета:</b> ознайомлення з основними елементами мови Visual Prolog; отримання навичок практичного використання фактів і правил у Visual Prolog.</p> <p><b>Завдання:</b> відповідно до завдання розробити текст програми з використанням фактів і правил; записати код у середовищі Visual Prolog та перевірити працездатність програми.</p>	<p align="center">2</p> <p align="center">10</p> <p align="center">2</p>	<p align="center">3</p> <p align="center">3</p> <p align="center">4</p>
<p align="center"><b>Тема 11. Структура програми Visual Prolog</b> <i>План лекції</i></p> <ol style="list-style-type: none"> <li>1. Основні розділи програми Visual Prolog.</li> <li>2. Директиви компілятора.</li> <li>3. Призначення розділів Visual Prolog.</li> <li>4. Семантика програм Visual Prolog.</li> </ol> <p align="center"><b>Список рекомендованих джерел:</b> Основний: 3 Додатковий: 8, 9 Інтернет-ресурси: 13</p> <p><b>Самостійна робота:</b> практичне ознайомлення з розділами Visual Prolog: CONSTANTS – опис констант; DOMAINS – опис доменів; FACTS – опис предикатів внутрішньої БД; PREDICATES – опис предикатів; CLAUSES – опис тверджень; GOAL – опис внутрішньої цілі; підготовка до лабораторного заняття № 11.</p>	<p align="center">2</p> <p align="center">8</p>	<p align="center">3</p>

Навчальна діяльність*	Роб. час студ., год.	Оцінювання, бали
<p align="center"><b>Лабораторне заняття № 11</b> <b>Повторювальні та рекурсивні обчислення</b></p> <p><b>Мета:</b> ознайомлення на практиці з основними розділами програми Visual Prolog; отримання навичок практичного застосування рекурсивних обчислень у Visual Prolog.</p> <p><b>Завдання:</b> відповідно до варіанту розробити текст програми з використанням повторювальних та рекурсивних обчислень; записати код та перевірити працездатність програми у середовищі Visual Prolog.</p>	2	4
<p align="center"><b>Тема 12. Предикати у Visual Prolog</b> <i>План лекції</i></p> <ol style="list-style-type: none"> <li>1. Визначення предикатів.</li> <li>2. Модулі та їх структура.</li> <li>3. Відсікання та заперечення.</li> <li>4. Структура та реалізація циклу з відкатом.</li> <li>5. Структура та реалізація рекурсії.</li> <li>6. Предикати другого порядку та анонімні предикати.</li> </ol> <p align="center"><b>Список рекомендованих джерел:</b></p> <p>Основний: 3 Додатковий: 8, 9 Інтернет-ресурси: 13</p> <p><b>Самостійна робота:</b> практичне засвоєння понять модуля, структури та предикатів у Visual Prolog; вивчення принципів роботи відсікання: область видимості відсікання, зелені та червоні відсікання, динамічне відсікання; підготовка до лабораторного заняття № 12.</p>	2	
<p align="center"><b>Лабораторне заняття № 12</b> <b>Застосування відсікання</b></p> <p><b>Мета:</b> ознайомлення з основними структурами та предикатами другого порядку; отримання навичок практичного застосування відсікання та заперечення у Visual Prolog.</p> <p><b>Завдання:</b> відповідно до завдання розробити текст програми з використанням відсікання та заперечення; записати код програми та перевірити працездатність у середовищі Visual Prolog.</p>	8	3
<p align="center"><b>Лабораторне заняття № 12</b> <b>Застосування відсікання</b></p> <p><b>Мета:</b> ознайомлення з основними структурами та предикатами другого порядку; отримання навичок практичного застосування відсікання та заперечення у Visual Prolog.</p> <p><b>Завдання:</b> відповідно до завдання розробити текст програми з використанням відсікання та заперечення; записати код програми та перевірити працездатність у середовищі Visual Prolog.</p>	2	5



Навчальна діяльність*	Роб. час студ., год.	Оцінювання, бали
<p><b>Тема 14. Створення додатків у середовищі Visual Prolog</b> <i>План лекції</i></p> <ol style="list-style-type: none"> <li>1. Створення консольних додатків.</li> <li>2. Створення проекту.</li> <li>3. Введення основного коду в програму.</li> <li>4. Створення додатків за допомогою графічного інтерфейсу.</li> <li>5. Управління за допомогою елементів форми.</li> </ol>	2	
<p style="text-align: center;"><b>Список рекомендованих джерел:</b></p> <p>Основний: 3 Додатковий: 8, 9, 10</p> <p><b>Самостійна робота:</b> ознайомлення з елементами форми, технологіями створення програми в режимі роботи з графічним інтерфейсом та способами управління у Visual Prolog; підготовка до лабораторного заняття № 14; підготовка до тестування.</p>	8	3
<p style="text-align: center;"><b>Лабораторне заняття № 14</b> <b>Експертні системи у середовищі Visual Prolog</b></p> <p><b>Мета:</b> ознайомлення з практичним використанням графічного інтерфейсу Visual Prolog та застосуванням елементів форми для створення додатків.</p> <p><b>Завдання:</b> відповідно до завдання та варіанту створити додаток у середовищі Visual Prolog за допомогою графічного інтерфейсу та елементів форми.</p>	2	5
<b>Всього</b>	<b>180</b>	<b>100</b>

\* всі лабораторні завдання виконуються на основі інтерактивних методів навчання у комп'ютерному середовищі

## **6. Список рекомендованих джерел:**

### **Основний:**

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.
3. Шумейко О.О. Visual Prolog. Опануй на прикладах : навч. посіб. / О.О. Шумейко, В.М. Кнуренко. – Дніпропетровськ : Біла К.О., 2014. – 404 с.

### **Додатковий:**

4. Кургаєв О.П. Методи та системи штучного інтелекту : конспект лекцій для студентів напряму підготовки 6.050101 «Комп'ютерні науки» денної та заочної форми навчання / О.П. Кургаєв – К. : НУХТ, 2014. – 279 с.
5. Кургаєв О.П. Методи та системи штучного інтелекту : лабораторний практикум для студентів напряму підготовки 6.050101 «Комп'ютерні науки» денної та заочної форми навчання / О.П. Кургаєв – К. : НУХТ, 2015. – 219 с.
6. Кобець О.В. Методичні вказівки до виконання практичних робіт «Програмування на AutoLISP» з курсу «Основи програмування інженерних систем графіки» для студентів машинобудівних спеціальностей «Технологія машинобудування», «Інструментальне виробництво» денної та заочної форм навчання / О.В.Кобець. – Харків: НТУ «ХП», 2004. – 55 с.
7. Шекета В.І. Логічне програмування : метод. вказівки / В. І. Шекета. – Івано-Франківськ : ІФНТУНГ, 2009. – 91 с.
8. Марков В.Н. Современное логическое программирование на языке Visual Prolog 7.5: учебник. / В.Н. Марков – СПб.: БХВ-Петербург, 2016. – 544 с.
9. Юрчишин В.М. Пролог – мова логічного програмування : конспект лекцій / В. М. Юрчишин, В. І. Шекета, Л. М. Гобир. – Івано-Франківськ : ІФНТУНГ, 2005. – 138 с.
10. Eduardo Costa. Visual Prolog 7.3 for Tyros. / Eduardo Costa. – New York: Springer-Verlag, 2010. – 270 p.

### **Інтернет-ресурси:**

11. Кристиан Кеннек. Інтерпретація Лиспа и Scheme. URL: <http://blog.ilammy.net/lisp/>.
12. Д. Троицкий. Программирование на AutoLISP. URL: <http://www.cad.dp.ua/kurs/index.html#CONTENTS>.
13. О.П. Солдатова, И.В. Лёзина. Логическое программирование на языке Visual Prolog. URL: <https://epdf.tips/-visual-prolog-.html>.



**7. Контроль та оцінювання результатів навчання:** положення про оцінювання результатів навчання студентів і аспірантів наказ КНТЕУ № 2891 від 16.09.2019 р.

Під час вивчення дисципліни «Функціональне та логічне програмування» викладачем здійснюється поточний та підсумковий контроль. Поточний контроль та оцінювання передбачає:

- перевірку рівня засвоєння теоретичного матеріалу (тестування за матеріалами лекції, який здійснюється на початку кожної наступної лекції);
- захист лабораторних робіт (проходить під час наступної лабораторної роботи);
- перевірка засвоєння матеріалу, що винесений на самостійне опрацювання під час фронтального опитування на лекції.

**8. Політика навчальної дисципліни:**

**8.1. Відвідування лекційних та лабораторних занять:** відвідування лекційних та лабораторних занять є обов'язковим. Допускаються пропуски занять з таких поважних причин, як хвороба (викладачу надається копія довідки від медичного закладу), участь в олімпіаді, творчому конкурсі тощо за попередньою домовленістю та згодою викладача за умови дозволу деканату (надаються документи чи інші матеріали, які підтверджують заявлену участь у діяльності студента).

**8.2. Відпрацювання пропущених занять:** відпрацювання пропущених занять є обов'язковим незалежно від причини пропущеного заняття. Лекційне заняття має бути відпрацьоване до наступної лекції на консультації викладача з використанням ПЗ 365 Office Teams. Відпрацювання лекційного матеріалу передбачає вивчення пропущеного теоретичного матеріалу та складання тесту за цим матеріалом. Лабораторне заняття відпрацьовується під час консультації викладача (розклад консультацій на сайті).

**8.3. Правила поведінки під час занять:** обов'язковим є дотримання техніки безпеки в комп'ютерних лабораторіях. Студенти повинні приймати активну участь в обговоренні навчального матеріалу, ознайомившись з ним напередодні (навчальний матеріал надається викладачем).

**8.4. За порушення академічної доброчесності** студенти будуть притягнені до академічної відповідальності у відповідності до положення про дотримання академічної доброчесності педагогічними, науково-педагогічними, науковими працівниками та здобувачами вищої освіти КНТЕУ (Наказ КНТЕУ від 02.02.2018 р. №377. Електронний ресурс. URL:

<https://knute.edu.ua/file/MTEyNDI=/f78c64a74cbbe5b4238729782d707efa.pdf>