Table of Contents

**ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ**

**ЗАТВЕРДЖУЮ**

|  |  |
| --- | --- |
| Декан факультету інформаційних технологій |  |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | **Тетяна ГОВОРУЩЕНКО** |
| Підпис | Ім’я, ПРІЗВИЩЕ |

2025 р.

### РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Функційне програмування

Назва дисципліни

| **Призначення Робочої програми** | Для освітньої програми спеціальності “123 Комп’ютерна інженерія” |
| --- | --- |
| **Рівень вищої освіти** | Перший (бакалаврський) |
| **Мова навчання** | Українська |
| **Обсяг дисципліни, кредитів ЄКТС** | 8 |
| **Статус дисципліни** | Вибіркова |
| **Факультет** | Інформаційних технологій |
| **Кафедра** | Комп’ютерної інженерії та інформаційних систем |

| **Форма здобуття освіти** | **Обсяг дисципліни** |  | **Кількість годин** |  |  |  |  |  | **Форма семестрового контролю** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Кредити ЄКТС** | **Години** | **Аудиторні заняття** |  |  |  |  | **Самостійна робота (в т.ч. ІРС)** |  |
|  |  |  | **Разом** | **Лекції** | **Лабораторні роботи** | **Практичні заняття** | **Семінарські заняття** |  |  |
| Д | 8 | 240 | 96 | 32 | 32 | 32 |  | 144 | Залік |

Робоча програма складена на основі освітніх програм підготовки бакалавра та стандарту вищої освіти спеціальності.

|  |  |
| --- | --- |
| Робоча програма складена \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | **д.т.н., доц. Сергій ЛИСЕНКО** |
|  | Науковий ступінь, вчене звання, Ім’я, ПРІЗВИЩЕ |
| Підпис |  |

Схвалена на засіданні кафедри комп’ютерної інженерії та інформаційних систем

Протокол від **29.08** **2025** № **1**.

|  |  |
| --- | --- |
| Зав. кафедри комп’ютерної інженерії та інформаційних систем | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ **Ірина ЗАСОРНОВА** |
| Назва | Підпис            Ім’я, ПРІЗВИЩЕ |

Хмельницький 2025

### 3. Пояснювальна записка

Дисципліна «Функційне програмування» відноситься до циклу вибіркових дисциплін та є важливою для формування у студентів сучасного погляду на розробку програмного забезпечення. Курс знайомить з парадигмою, що наголошує на використанні чистих функцій та незмінних даних, що дозволяє створювати надійні, масштабовані та легкі для тестування програмні системи. Опанування курсу формує міцну основу для подальшої професійної спеціалізації у таких сферах, як розробка розподілених систем, аналіз великих даних та штучний інтелект. У процесі навчання здобувачі набувають важливих соціальних навичок (soft skills): здатність до абстрактного мислення, аналізу та синтезу, системного мислення та застосування творчих здібностей для формування нових ідей.

**Мета дисципліни.** Оволодіння студентами основними концепціями функційного програмування на прикладі мови F#, такими як незмінний стан, функції вищого порядку та чисті функції, а також їх застосуванням для вирішення прикладних задач із залученням методів штучного інтелекту.

**Предмет дисципліни.** Основи програмування мовою F# з використанням концепцій та парадигм функційного програмування, їх застосування при вирішенні прикладних задач, зокрема у сфері штучного інтелекту.

**Завдання дисципліни.** Формування практичних навичок програмування мовою F# із застосуванням функційної парадигми; навчання вибору засобів функційного програмування для розробки прикладних програмних систем; ознайомлення з технікою програмування задач штучного інтелекту з використанням функційного підходу.

**Результати навчання.** Після вивчення дисципліни студент повинен: **знати** теоретичні положення, що лежать в основі парадигм функційного програмування, основні концепції (незмінний стан, функції вищого порядку, чисті функції, лямбда-обчислення) та принципи програмування задач штучного інтелекту; **володіти** інструментами функційного програмування для реалізації складних алгоритмічних рішень; **вміти** застосовувати набуті знання для проєктування та реалізації програмних систем; здійснювати пошук, оброблення та аналіз інформації для розв’язання прикладних задач; поєднувати теорію і практику для прийняття оптимальних рішень; виконувати експериментальні дослідження та системно мислити для формування нових ідей; оцінювати отримані результати та аргументовано захищати прийняті рішення.

### 4. Структура залікових кредитів дисципліни

| **Назва розділу (теми)** | **лекції** | **лабораторні** | **практичні** | **СРС** |
| --- | --- | --- | --- | --- |
| Тема 1. Основи функційного програмування та F# | 8 | 8 | 8 | 34 |
| Тема 2. Робота з колекціями та структурами даних в F# | 8 | 8 | 8 | 38 |
| Тема 3. Поглиблені засоби функційного програмування | 8 | 8 | 8 | 38 |
| Тема 4. Застосування ФП в машинному навчанні | 8 | 8 | 8 | 34 |
| **Разом за семестр:** | **32** | **32** | **32** | **144** |

### 5. Програма навчальної дисципліни

#### 5.1 Зміст лекційного курсу

| **Номер лекції** | **Перелік тем лекцій, їхні анотації** | **Кількість годин** |
| --- | --- | --- |
|  | **Тема 1. Основи функційного програмування та F#** | **8** |
| 1 | Вступ до функційного програмування. Парадигми програмування. Основні принципи ФП: чисті функції, незмінність даних, відсутність побічних ефектів. Переваги функційного підходу.Літ.: [8] | 2 |
| 2 | Основи мови F#. Синтаксис, типи даних, оператори. Робота з модулями та просторами імен. Виведення типів. Створення першої програми.Літ.: [2, 8] | 2 |
| 3 | Функції у F#. Оголошення функцій. Рекурсія та хвостова рекурсія. Каррінг та часткове застосування функцій. Лямбда-вирази.Літ.: [3, 5] | 2 |
| 4 | Системи типів та зіставлення зі зразком. Алгебраїчні типи даних: записи та розмічені об’єднання. Потужність зіставлення зі зразком для обробки даних.Літ.: [4, 8] | 2 |
|  | **Тема 2. Робота з колекціями та структурами даних в F#** | **8** |
| 5 | Списки та послідовності. Робота зі списками (List) як основною незмінною колекцією. Ліниві обчислення за допомогою послідовностей (Seq).Літ.: [3, 8] | 2 |
| 6 | Масиви та інші колекції. Робота зі змінними масивами (Array). Огляд інших колекцій .NET та їх використання у функційному стилі.Літ.: [3, 8] | 2 |
| 7 | Асоціативні колекції. Робота з незмінними Map та Set. Ефективні операції пошуку, додавання та видалення елементів.Літ.: [3, 8] | 2 |
| 8 | Проєктування функційних структур даних. Реалізація власних незмінних структур даних, таких як стек, черга або дерево.Літ.: [3, 7] | 2 |
|  | **Тема 3. Поглиблені засоби функційного програмування** | **8** |
| 9 | Функції вищого порядку. Використання функцій як аргументів та результатів інших функцій. Функції map, filter, fold для роботи з колекціями.Літ.: [5, 8] | 2 |
| 10 | Композиція функцій та конвеєризація. Побудова складних операцій за допомогою операторів композиції (>>) та конвеєризації (|>).Літ.: [5, 8] | 2 |
| 11 | Асинхронне та паралельне програмування. Асинхронні робочі процеси (Async). Основи паралельної обробки даних у функційному стилі.Літ.: [2, 8] | 2 |
| 12 | Обробка помилок та тип Option. Функційний підхід до обробки помилок без винятків. Використання типу Option для представлення відсутності значення.Літ.: [3, 8] | 2 |
|  | **Тема 4. Застосування ФП в машинному навчанні** | **8** |
| 13 | Вступ до машинного навчання з F#. Огляд бібліотек для МН (ML.NET, Accord.NET). Переваги ФП для задач аналізу даних.Літ.: [1, 5, 6] | 2 |
| 14 | Підготовка та обробка даних. Використання функційних підходів для очищення, трансформації та нормалізації даних.Літ.: [1, 2, 4] | 2 |
| 15 | Побудова моделей машинного навчання. Реалізація простих моделей (напр., лінійна регресія) у функційному стилі. Робота з API бібліотек.Літ.: [1, 6, 7] | 2 |
| 16 | Оцінка та валідація моделей. Розробка функцій для оцінки якості моделей. Застосування конвеєрів для створення повного циклу МН.Літ.: [1, 4, 7] | 2 |
|  | **Разом за семестр:** | **32** |

#### 5.2 Зміст практичних занять

| **№ п/п** | **Тема практичного заняття** | **Кількість годин** |
| --- | --- | --- |
| 1 | Налаштування середовища розробки. Перші кроки в F#: F# Interactive, створення проєктів. | 2 |
| 2 | Робота з базовими типами та виразами. Розв’язання задач на умовні конструкції та цикли. | 2 |
| 3 | Практикум з рекурсії. Розв’язання класичних задач (факторіал, числа Фібоначчі) з хвостовою рекурсією. | 2 |
| 4 | Моделювання доменів за допомогою алгебраїчних типів даних. | 2 |
| 5 | Маніпуляції зі списками: базові операції, рекурсивна обробка. | 2 |
| 6 | Ефективна робота з послідовностями для обробки великих обсягів даних. | 2 |
| 7 | Використання Map та Set для вирішення задач, що потребують унікальності або пошуку за ключем. | 2 |
| 8 | Розробка власної незмінної структури даних (напр., бінарне дерево). | 2 |
| 9 | Практичне застосування функцій map, filter, fold для трансформації даних. | 2 |
| 10 | Побудова конвеєрів обробки даних за допомогою оператора |>. | 2 |
| 11 | Розв’язання задач на асинхронну обробку даних (напр., завантаження даних з мережі). | 2 |
| 12 | Рефакторинг коду з використанням типу Option для безпечної обробки відсутніх значень. | 2 |
| 13 | Знайомство з датафреймами та бібліотеками для роботи з даними в F#. | 2 |
| 14 | Виконання завдань з попередньої обробки реального набору даних. | 2 |
| 15 | Тренування простої моделі класифікації або регресії за допомогою ML.NET. | 2 |
| 16 | Написання функцій для розрахунку метрик якості моделі (точність, повнота). | 2 |
|  | **Разом за семестр:** | **32** |

#### 5.3 Зміст лабораторних занять

| **№ п/п** | **Тема лабораторного заняття** | **Кількість годин** |
| --- | --- | --- |
| 1 | Розробка базових програм та функцій на F#. Реалізація набору функцій для математичних обчислень з використанням рекурсії та зіставлення зі зразком. | 4 |
| 2 | Робота зі списками та послідовностями. Створення програми для аналізу текстових даних (підрахунок частоти слів, пошук патернів) за допомогою функцій вищого порядку. | 4 |
| 3 | Реалізація та використання власних структур даних. Розробка модуля, що реалізує незмінне бінарне дерево пошуку з операціями додавання, пошуку та видалення. | 4 |
| 4 | Поглиблене маніпулювання функціями. Створення бібліотеки функцій для обробки даних з активним використанням композиції, каррінгу та конвеєризації. | 4 |
| 5 | Розробка асинхронних та паралельних програм. Створення програми, яка паралельно обробляє колекцію файлів або асинхронно завантажує дані з кількох веб-ресурсів. | 4 |
| 6 | Функційна обробка та аналіз даних. Розробка конвеєру для підготовки табличних даних до машинного навчання: завантаження, очищення, трансформація ознак. | 4 |
| 7 | Реалізація простого алгоритму машинного навчання. Написання з нуля алгоритму k-найближчих сусідів (k-NN) у функційному стилі для задачі класифікації. | 4 |
| 8 | Побудова та оцінка моделі машинного навчання. Розробка повноцінного додатку, що тренує модель на основі бібліотеки ML.NET, оцінює її якість та зберігає для подальшого використання. | 4 |
|  | **Разом за семестр:** | **32** |

#### 5.4 Зміст самостійної роботи здобувача вищої освіти

Самостійна робота студентів полягає у систематичному опрацюванні програмного матеріалу з відповідних джерел інформації, підготовці до практичних та лабораторних занять, виконанні індивідуальних домашніх завдань (ІДЗ). До послуг студентів сторінка кафедри у Модульному середовищі для навчання, де розміщені Робоча програма дисципліни та необхідні матеріали з її навчально-методичного забезпечення та контролю результатів навчання.

| **Номер тижня** | **Вид самостійної роботи** | **Кількість годин** |
| --- | --- | --- |
| 1-2 | Опрацювання лекційного матеріалу. Підготовка до виконання та захисту ЛР №1 та підготовка до ПЗ №1-4. | 18 |
| 3-4 | Опрацювання лекційного матеріалу. Підготовка до виконання та захисту ЛР №2 та підготовка до ПЗ №5-6. Виконання ІДЗ №1 “Реалізація парсера для простої мови розмітки”. | 20 |
| 5-6 | Опрацювання лекційного матеріалу. Підготовка до виконання та захисту ЛР №3 та підготовка до ПЗ №7-8. | 16 |
| 7-8 | Опрацювання лекційного матеріалу. Підготовка до виконання та захисту ЛР №4 та підготовка до ПЗ №9–10. Виконання ІДЗ №2 “Створення функційної бібліотеки для роботи з геометричними фігурами”. | 20 |
| 9-10 | Опрацювання лекційного матеріалу. Підготовка до виконання та захисту ЛР №5 та підготовка до ПЗ №11-12. | 16 |
| 11-12 | Опрацювання лекційного матеріалу. Підготовка до виконання та захисту ЛР №6 та підготовка до ПЗ №13-14. Виконання ІДЗ №3 “Розробка паралельного веб-сканера для збору інформації”. | 20 |
| 13-14 | Опрацювання лекційного матеріалу. Підготовка до виконання та захисту ЛР №7 та підготовка до ПЗ №15-16. | 16 |
| 15-17 | Опрацювання лекційного матеріалу. Підготовка до виконання та захисту ЛР №8. Виконання ІДЗ №4 “Побудова та аналіз моделі для прогнозування на основі реального набору даних”. Захист ІДЗ. | 18 |
|  | **Разом за семестр:** | **144** |

**Примітки:** ЛР – лабораторна робота, ПЗ – практичне заняття, ІДЗ – індивідуальне домашнє завдання.

### 6. Технології та методи навчання

Процес навчання з дисципліни ґрунтується на використанні традиційних та сучасних технологій та методів навчання, зокрема: **лекції** (з використанням проблемного й інтерактивного навчання, інформаційно-комунікаційних технологій); **практичні заняття** (з використанням інструктування, розв’язування типових і прикладних задач, аналізу кейсів); **лабораторні заняття** (з фокусом на програмній реалізації функційних концепцій, тестуванні їхньої ефективності); **самостійна робота** (опрацювання теоретичного матеріалу, підготовка до виконання лабораторних робіт, виконання та захист ІДЗ), з використанням інформаційно-комп’ютерних технологій та технологій дистанційного навчання.

### 7. Методи контролю

Поточний контроль здійснюється під час аудиторних практичних та лабораторних занять, а також у дні проведення контрольних заходів, встановлених робочою програмою і графіком освітнього процесу, зокрема, з використанням Модульного середовища для навчання. Водночас використовуються такі методи поточного контролю:

* оцінювання результатів роботи на практичних заняттях (опитування, розв’язування задач);
* оцінювання результатів захисту лабораторних робіт;
* захист та оцінювання результатів виконання ІДЗ;

Підсумкова семестрова оцінка виставляється за результатами поточного контролю. Здобувач вищої освіти, який набрав з будь-якого виду навчальної роботи, суму балів нижчу за 60 відсотків від максимального балу, вважається таким, який має академічну заборгованість.

### 8. Політика дисципліни

Політика навчальної дисципліни загалом визначається системою вимог до здобувача вищої освіти. Відвідування занять з дисципліни є обов’язковим. Успішне опанування дисципліни передбачає необхідність підготовки до практичних та лабораторних занять, активну роботу на занятті, участь у дискусіях та своєчасне виконання ІДЗ.

Здобувачі вищої освіти мають дотримуватися встановлених термінів виконання всіх видів навчальної роботи. Пропущене заняття студент зобов’язаний відпрацювати у встановлений викладачем термін.

Здобувач вищої освіти, виконуючи самостійну роботу, має дотримуватися політики доброчесності. У разі виявлення порушення політики академічної доброчесності (плагіат, списування) здобувач отримує незадовільну оцінку і має повторно виконати завдання. Будь-які форми порушення академічної доброчесності **не допускаються**.

### 9. Оцінювання результатів навчання студентів у семестрі

Оцінювання академічних досягнень здобувача вищої освіти здійснюється, відповідно до «Положення про контроль і оцінювання результатів навчання здобувачів вищої освіти у ХНУ». При поточному оцінюванні кожна структурна одиниця навчальної роботи може бути зарахована, якщо здобувач набрав не менше 60 відсотків від максимально можливої суми балів.

**Таблиця - Критерії оцінювання навчальних досягнень здобувача вищої освіти**

| **Оцінка та рівень досягнення здобувачем запланованих ПРН та сформованих компетентностей** | **Узагальнений зміст критерія оцінювання** |
| --- | --- |
| **Відмінно (високий)** | Здобувач вищої освіти глибоко і у повному обсязі опанував зміст навчального матеріалу, легко в ньому орієнтується і вміло використовує понятійний апарат; уміє пов’язувати теорію з практикою, вирішувати практичні завдання, впевнено висловлювати і обґрунтовувати свої судження. |
| **Добре (середній)** | Здобувач вищої освіти виявив повне засвоєння навчального матеріалу, володіє понятійним апаратом, орієнтується у вивченому матеріалі; свідомо використовує теоретичні знання для вирішення практичних задач; у змісті і формі відповіді можуть мати місце окремі неточності. |
| **Задовільно (достатній)** | Здобувач вищої освіти виявив знання основного програмного матеріалу в обсязі, необхідному для подальшого навчання, справляється з виконанням практичних завдань, передбачених програмою. Допускає неточності і суттєві помилки у відповіді. |
| **Незадовільно (недостатній)** | Здобувач вищої освіти виявив розрізнені, безсистемні знання, допускається помилок у визначенні понять, хаотично і невпевнено викладає матеріал, не може використовувати знання при вирішенні практичних завдань. |

**Структурування дисципліни за видами навчальної роботи і оцінювання результатів навчання**

| **Види навчальної роботи** | **Кількість балів (min-max)** | **Загальна кількість балів** |
| --- | --- | --- |
| Лабораторні роботи (8 робіт) | 24-40 | 40 |
| Практичні заняття (активність, виконання завдань) | 12-20 | 20 |
| Індивідуальні домашні завдання (4 завдання) | 24-40 | 40 |
| **Всього за семестр** | **60-100** | **100** |

**Оцінювання результатів захисту лабораторної роботи**

Кожна з 8 лабораторних робіт оцінюється за шкалою від 3 до 5 балів. Враховується самостійність та правильність програмної реалізації, повнота відповіді та знання теоретичних основ, якість програмного коду.

**Оцінювання на практичних заняттях**

Загальна оцінка за практичні заняття (до 20 балів) складається з оцінювання активності, усних відповідей та виконання завдань протягом семестру.

**Оцінювання результатів виконання індивідуального домашнього завдання**

Кожне з 4 ІДЗ оцінюється за шкалою від 6 до 10 балів. Враховується самостійність виконання, правильність розв’язання поставлених задач, обґрунтованість вибору методів, повнота пояснень та якість оформлення звіту.

**Підсумкова семестрова оцінка**

Підсумкова семестрова оцінка визначається в автоматизованому режимі після внесення викладачем результатів оцінювання до електронного журналу.

**Таблиця – Співвідношення інституційної шкали оцінювання і шкали оцінювання ЄКТС**

| **Оцінка ЄКТС** | **Рейтингова шкала балів** | **Інституційна оцінка** |
| --- | --- | --- |
| A | 90-100 | **Зараховано** |
| B | 83-89 |  |
| C | 73-82 |  |
| D | 66-72 |  |
| E | 60-65 |  |
| FX | 40-59 | **Незараховано** |
| F | 0-39 |  |

Семестровий залік виставляється на останньому занятті за умови, якщо загальна сума балів, яку накопичив здобувач, перебуває у межах від 60 до 100 балів.

### 10. Питання для самоконтролю результатів навчання

1. Що таке функційна парадигма програмування?
2. Ключові принципи ФП: чисті функції, незмінність.
3. Основи синтаксису F#: оголошення змінних, функції.
4. Що таке виведення типів і як воно працює в F#?
5. Поняття рекурсії. Чим хвостова рекурсія відрізняється від звичайної?
6. Що таке каррінг та часткове застосування функцій?
7. Призначення та використання лямбда-виразів.
8. Що таке алгебраїчні типи даних? Наведіть приклади записів та розмічених об’єднань.
9. Як працює зіставлення зі зразком в F#?
10. Основні відмінності між списками (List) та послідовностями (Seq).
11. Що таке ліниві обчислення і де вони застосовуються?
12. Як працювати з масивами в F#?
13. Призначення та переваги незмінних Map та Set.
14. Як спроєктувати власну незмінну структуру даних?
15. Що таке функція вищого порядку? Наведіть приклади.
16. Призначення функцій map, filter, fold.
17. Що таке композиція функцій?
18. Як оператор конвеєризації (|>) покращує читабельність коду?
19. Основи асинхронного програмування в F#.
20. Як уникнути винятків за допомогою типу Option?
21. Які бібліотеки для машинного навчання можна використовувати з F#?
22. Переваги ФП для задач підготовки та обробки даних.
23. Як виглядає конвеєр обробки даних у функційному стилі?
24. Основні етапи побудови моделі машинного навчання.
25. Як оцінити якість моделі класифікації або регресії?

### 11. Навчально-методичне забезпечення

Освітній процес з дисципліни «Функційне програмування» повністю і в достатній кількості забезпечений необхідною навчально-методичною літературою, перелік якої наведено у розділі 13.

### 12. Матеріально-технічне та програмне забезпечення дисципліни

Для проведення лекційних, практичних та лабораторних занять використовуються спеціалізовані комп’ютерні класи кафедри, оснащені сучасною комп’ютерною технікою та мультимедійним обладнанням. Програмне забезпечення включає операційні системи (Windows, Linux), середовища розробки (Visual Studio, Visual Studio Code з плагіном Ionide, JetBrains Rider), .NET SDK, системи контролю версій.

### 13. Рекомендована література

1. Bishop, C. M. Pattern Recognition and Machine Learning / C. M. Bishop. – Springer, 2018. – 738 р.
2. Dixon, J. Mastering .NET Machine Learning / J. Dixon. – Packt Publishing Ltd, 2019. – 358 р.
3. Masood, A. Learning F# Functional Data Structures and Algorithms. / A. Masood. – Packt Publishing Ltd, 2020. – 206 р.
4. Mohri, M., Rostamizadeh, A., Talwalkar A. Foundations of Machine Learning / M. Mohri, A. Rostamizadeh, A. Talwalkar. - The MIT Press, 2020. – 432 р.
5. Mukherjee, S. F# for Machine Learning Essentials / S. Mukherjee. – Packt Publishing Ltd, 2018. – 194 р.
6. Murphy, K. P., Bach, F. Machine Learning: A Probabilistic Perspective / K. P. Murphy, F. Bach. – The MIT Press, 2019. – 1104 р.
7. Shalev-Shwartz, S., Ben-David, S. Understanding Machine Learning: From Theory to Algorithms / S. Shalev-Shwartz, S. Ben-David. – Cambridge University Press, 2020. – 449 р.
8. F# Language Reference [Microsoft, 2020]. Доступ до ресурсу: https://docs.microsoft.com/en-us/dotnet/fsharp/language-reference/
9. Модульне середовище для навчання MOODLE. Доступ до ресурсу: https://msn.khnu.km.ua/course/view.php?id=546
10. Електронна бібліотека університету. Доступ до ресурсу: http://lib.khnu.km.ua/asp/php\_f/plage\_lib.php

### 14. Інформаційні ресурси

1. Модульне середовище для навчання. URL: https://msn.khnu.km.ua/
2. Електронна бібліотека ХНУ. URL: http://library.khmnu.edu.ua/
3. Інституційний репозитарій ХНУ. URL: https://elar.khmnu.edu.ua/home

### АНОТАЦІЯ ДИСЦИПЛІНИ (ДЛЯ КАТАЛОГУ)

ФУНКЦІЙНЕ ПРОГРАМУВАННЯ

|  |  |
| --- | --- |
| **Тип дисципліни** | Вибіркова |
| **Рівень вищої освіти** | Перший (бакалаврський) |
| **Мова викладання** | Українська |
| **Семестр** | 6 |
| **Кількість призначених кредитів ЄКТС** | 8,0 |
| **Форми здобуття освіти, для яких викладається дисципліна** | Очна (денна) |

**Результати навчання.** Після вивчення дисципліни студент повинен: **знати** теоретичні положення та основні концепції функційного програмування (чисті функції, незмінність, функції вищого порядку, лямбда-обчислення) та принципи їх застосування в задачах штучного інтелекту; **володіти** інструментами мови F# для реалізації складних алгоритмів; **вміти** застосовувати набуті знання для проєктування надійних програмних систем, поєднувати теорію і практику для прийняття оптимальних рішень, виконувати експериментальні дослідження та аргументовано захищати отримані результати.

**Зміст навчальної дисципліни.** Вступ до функційної парадигми. Основи мови F#: синтаксис, система типів, функції, рекурсія, каррінг. Алгебраїчні типи даних та зіставлення зі зразком. Робота з незмінними колекціями: списки, послідовності, мапи. Функції вищого порядку (map, filter, fold). Композиція функцій та конвеєризація. Асинхронне та паралельне програмування. Функційні підходи до обробки помилок. Застосування F# для задач машинного навчання: підготовка даних, побудова та оцінка моделей з використанням відповідних бібліотек.

**Запланована навчальна діяльність.** Лекції – 32 год., лабораторні заняття – 32 год., практичні заняття – 32 год., самостійна робота – 144 год.; разом – 240 год.

**Форми (методи) навчання:** лекції (проблемні, інтерактивні); практичні заняття (аналіз кейсів, розв’язування задач); лабораторні заняття (програмна реалізація функційних концепцій); самостійна робота (виконання ІДЗ, опрацювання теоретичного матеріалу).

**Форми оцінювання результатів навчання:** оцінювання роботи на практичних заняттях; оцінювання виконання та захисту лабораторних робіт; оцінювання виконання та захисту індивідуальних домашніх завдань.

**Вид семестрового контролю:** залік.

**Навчальні ресурси:**

1. Masood, A. Learning F# Functional Data Structures and Algorithms. Packt Publishing Ltd, 2020.
2. Mukherjee, S. F# for Machine Learning Essentials. Packt Publishing Ltd, 2018.
3. F# Language Reference [Microsoft, 2020]. URL: https://docs.microsoft.com/en-us/dotnet/fsharp/language-reference/
4. Модульне середовище для навчання. URL: https://msn.khnu.km.ua/
5. Електронна бібліотека ХНУ. Доступ до рecypcy: http://library.khmnu.edu.ua/

**Викладачі:** д.т.н., доц. Лисенко С.М.