



**ДЕРЖАВНИЙ ТОРГОВЕЛЬНО-ЕКОНОМІЧНИЙ
УНІВЕРСИТЕТ**
Факультет інформаційних технологій
Кафедра комп'ютерних наук та інформаційних систем

СИЛАБУС (SYLLABUS)
Дисципліна «Крос-платформне програмування /
Cross-platform programming»

ІНФОРМАЦІЯ ПРО ВИКЛАДАЧА

Викладач	Базурін Віталій Миколайович
Науковий ступінь	Кандидат педагогічних наук
Вчене звання	Доцент
Посада	Доцент кафедри комп'ютерних наук та інформаційних систем
Адреса кафедри	м. Київ, вул. Кіото 19, каб. Б-507, Б-526
E-mail	compdep@knute.edu.ua
Консультації	Відповідно до графіку індивідуальних консультацій на сайті кафедри

ПОЛІТИКА АКАДЕМІЧНОЇ ДОБРОЧЕСНОСТІ

<https://knute.edu.ua/file/MzEyMQ==/c12a9f74e87d9154696ca0f761da2e5c.pdf>

Дотримання академічної доброчесності передбачає:

- самостійне виконання навчальних завдань, завдань поточного та підсумкового контролю результатів навчання (для осіб з особливими освітніми потребами ця вимога застосовується з урахуванням їхніх індивідуальних потреб і можливостей);
- посилання на джерела інформації у разі використання не авторських ідей, розробок, тверджень, відомостей і т.п.;
- дотримання норм законодавства про авторське право і суміжні права;
- надання достовірної інформації про результати власної наукової діяльності, використанні методики досліджень і джерела інформації.

Порушенням академічної доброчесності вважається:

- академічний плагіат – оприлюднення (частково або повністю) наукових (творчих) результатів, отриманих іншими особами, як результатів власного дослідження (творчості) та/або відтворення опублікованих текстів (оприлюднених творів мистецтва) інших авторів без зазначення авторства;
- самоплагіат – оприлюднення (частково або повністю) власних раніше опублікованих наукових результатів як нових наукових результатів;
- фабрикація – вигадкування даних чи фактів, що використовуються в наукових дослідженнях;
- фальсифікація – свідомо зміна чи модифікація вже наявних даних, що стосуються наукових досліджень.

За порушення академічної доброчесності здобувачі освіти можуть бути притягнені до академічної відповідальності:

- повторне проходження оцінювання (модульний контроль, іспит, залік тощо);
- повторне проходження відповідного освітнього компонента освітньо-професійної програми;
- відрахування з Університету;

- позбавлення наданих університетом пільг;
 - відмова у присудженні відповідного ступеня вищої освіти;
- ПОЛІТИКА ЩОДО ВІДВІДУВАННЯ ЗАНЯТЬ**
- відвідування занять є обов'язковим;
 - за об'єктивних причин (наприклад, хвороба, міжнародне стажування та ін.) навчання може відбуватись в он-лайн формі за погодженням із викладачем дисципліни.

ОПИС НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Назва дисципліни / тип дисципліни	Крос-платформне програмування / обов'язкова
Навчальний рік	2023-2024
Факультет	Факультет інформаційних технологій
Курс	4
Семестр	7,8
Освітній ступінь	Бакалавр
Галузь знань	12 «Інформаційні технології»
Спеціальність	126 «Інформаційні системи та технології»
Загальна характеристика	Кількість годин – 270 Кількість кредитів – 9 Види занять: лекції, лабораторні, самостійна робота. Співвідношення аудиторних годин і годин самостійної роботи - 110/160 Мова викладання – українська Форма викладання – очна
Підсумковий контроль	Екзамен
Програмне забезпечення	Visual Studio, Android Studio
Обладнання	Проектор, комп'ютерна техніка із встановленим програмним забезпеченням та доступом до мережі Інтернет.
Необхідні попередні дисципліни	«Основи теорії інформаційних систем», «Алгоритмізація та програмування»
Методика вивчення	Методика вивчення дисципліни полягає у набутті студентами знань теоретичного і практично-прикладного характеру під час лекцій, лабораторних занять, самостійної роботи та вивчення першоджерел і навчально-методичної літератури.
Мета і завдання	<i>Метою</i> вивчення дисципліни «Крос-платформне програмування» є вивчення теоретичних основ програмування, засвоєння основних прийомів програмування з урахуванням сучасних концепцій і тенденцій розвитку технологій програмування, забезпечення навичок розв'язування реальних науково-технічних задач різної складності за допомогою персонального комп'ютера, набуття студентами знань та умінь для створення крос-платформних програмних продуктів на базі відповідних вивчених технологій та засвоєння необхідних знань щодо технологій створення додатків, які базуються на сучасних мобільних платформах і техніки їх застосування у реалізації бізнес-комунікацій. <i>Завданням</i> вивчення дисципліни «Крос-платформне програмування» є оволодіння принципами використання засобів крос-платформного

	програмування; стандартами та технологіями взаємодії, застосування, використання даних, інформації та знань в організаціях та бізнес-діяльності на основі мобільних пристроїв.
Місце дисципліни в освітньо-професійній програмі	
Загальні компетентності	КЗ 2 Здатність застосовувати знання у практичних ситуаціях. КЗ 3 Здатність до розуміння предметної області та професійної діяльності.
Фахові компетентності (результати навчання)	КС 2 Здатність застосовувати стандарти в області інформаційних систем та технологій при розробці функціональних профілів, побудові та інтеграції систем, продуктів, сервісів і елементів інфраструктури організації. КС 3 Здатність до проектування, розробки, налагодження та вдосконалення системного, комунікаційного та програмно-апаратного забезпечення інформаційних систем та технологій, Інтернету речей (IoT), комп'ютерно-інтегрованих систем та системної мережної структури, управління ними. КС 4. Здатність проектувати, розробляти та використовувати засоби реалізації інформаційних систем, технологій та інфокомунікацій (методичні, інформаційні, алгоритмічні, технічні, програмні та інші). КС 6 Здатність використовувати сучасні інформаційні системи та технології (виробничі, підтримки прийняття рішень, інтелектуального аналізу даних та інші), методики й техніки кібербезпеки під час виконання функціональних завдань та обов'язків. КС 12 Здатність управляти та користуватися сучасними інформаційно-комунікаційними системами та технологіями (у тому числі такими, що базуються на використанні Інтернет).
Програмні результати навчання	ПР 3 Використовувати базові знання інформатики й сучасних інформаційних систем та технологій, навички програмування, технології безпечної роботи в комп'ютерних мережах, методи створення баз даних та інтернет-ресурсів, технології розроблення алгоритмів і комп'ютерних програм мовами високого рівня із застосуванням об'єктно-орієнтованого програмування для розв'язання задач проектування і використання інформаційних систем та технологій. ПР 5 Аргументувати вибір програмних та технічних засобів для створення інформаційних систем та технологій на основі аналізу їх властивостей, призначення і технічних характеристик з урахуванням вимог до системи і експлуатаційних умов; мати навички налагодження та тестування програмних і технічних засобів інформаційних систем та технологій. ПР 6 Демонструвати знання сучасного рівня технологій інформаційних систем, практичні навички програмування та використання прикладних і спеціалізованих комп'ютерних систем та середовищ з метою їх запровадження у професійній діяльності. ПР 7 Обґрунтовувати вибір технічної структури та розробляти відповідне програмне забезпечення, що входить до складу інформаційних систем та технологій.

ТЕМАТИКА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Тема 1. Платформа .Net. Середовище розробки Visual Studio.

Історія розроблення Visual Studio.Net. .Net Framework. Основні терміни і визначення (MSIL, CLR, CLS, Assembly, Assembly manifest, Jit, Linking). Вікна VS (вікно текстового редактора, вікна

solution explorer, class view, error list, toolbox). Меню VS. Консольні проекти та проекти Windows Forms.

Тема 2. Основні поняття мови програмування C#. Типи даних у C#.

Змінні. Прості типи даних. Літерали. Вирази (арифметичні, логічні). Операції, пріоритет їхнього виконання. Область видимості, простір імен. Неявне перетворення типів. Явне перетворення типів. Складні типи – класи, перечислення (enum), структури, масиви. Робота із рядками символів (тип string).

Тема 3. Створення проектів на C# та програмування алгоритмів, функції, Windows Forms – проекти.

Створення консольного проекту. Розгалуження, оператори `?`, `if`, `switch`, тернарні операції. Оператори циклів: `do`, `for`, `while`, `foreach`, оператор `break`. Визначення, синтаксис, тип значення функції, що повертається. Параметри функцій – позиційні, іменовані; атрибути `ref`, `out`. Область видимості змінних у функціях. Перевантаження функцій. Поняття делегата. Створення проекту Windows Forms. Режим Design для форми. Розроблення інтерфейсу програми. Засоби для дизайну форми. Toolbox, контролзи, вікно властивостей, вікна solution explorer та class view. Найчастіше вживані контролзи: `TextBox`, `Label`, `Button`, `ListBox`, `ListView`, `ComboBox`, `Panel`, `RadioButton`, `Table`, `DataGridView`. Створення меню – контролз `ToolStrip`. Створення локальних звітів. Клас `ReportViewer`.

Тема 4. Основи крос-платформного програмування на C#.

Кросплатформне програмування сутності і поняття. Клас, об'єкт, екземпляр класу. Метод. Властивість. Інкапсуляція, успадкування, поліморфізм. Конструктор, деструктор. Змінні за значеннями та за посиланнями. Упакування, розпакування. Купа, стек. Модифікатори доступу, атрибути. Атрибути `virtual`, `override`, `new`, `abstract`, `base`, `static`, `sealed`. Індексатори. Керування пам'яттю. Збирач сміття `garbage collector`. Поняття `Finalizer` і `Dispose`. Інтерфейси. Специфікація інтерфейсу як контракту. Перевантаження операторів у класах. Методи `explicit` та `implicit`. Делегат для створення подій. Джерела подій та приймачі подій. Детальний розгляд простих програм з подіями.

Тема 5. Динамічні структури даних.

Простір імен `System.Collections`. Колекції. Інтерфейси `IList`, `ICollection`, `IDictionary`, `IDisposable`. Клас `ArrayList`. Основні властивості колекцій: `Capacity`, `Count`, `IsFixedSize`, `IsReadOnly`. Основні методи колекцій: `Add`, `AddRange`, `BinarySearch`, `Clear`, `Clone`, `IndexOf`, `Insert`, `InsertRange`, `Remove`, `Sort`, `ToArray`, `ToString`. Інтерфейси `IComparable` та `IComparer`. Сортування та пошук у колекціях типу `ArrayList` та типу `DictionaryBase`. Узагальнені колекції. Простір імен `System.Collections.Generic`. Тип `<List>`. Основні методи, сортування та пошук. Графи та дерева. Бінарне дерево та префіксне дерево. Рекурсивні методи для роботи з деревами.

Тема 6. Програмування з використанням компонент.

Поняття компоненти. Вибір між компонентою, керуючим елементом та класом. Інтерфейс `IComponent` та клас `System.ComponentModel.Component`. Члени інтерфейсу `IComponent`. Розміщення компоненти у контейнері. Маршалінг компонент. Стратегії інтеграції програмного забезпечення. Розподілена архітектура компонентних систем. Формальні та візуальні методи конструювання компонент.

Тема 7. Елементи керування.

Поняття елемента керування. Класи `System.Windows.Forms.Control` і `System.Web.UI.Control`. Поведінка у режимі дизайну та у режимі виконання. Використання атрибутів. Підтримка керуючого елемента редактором форм. Конвертори типів. Розробка та збирання елемента Розподілена архітектура компонентних систем. Компоненто орієнтоване проектування. Добирання застосування сервісів, компонентів, протоколів зв'язку. Брокери

об'єктних запитів. Монітори оброблення транзакцій. Виклики віддалених процедур. Вибір та застосування сервісів, компонент і протоколів.

Тема 8. Основи розробки і побудови мобільних додатків.

Вступ до Android. Мови програмування для Android. Перший додаток для Android. Структура проекту. Запуск проекту. Система контролю версій. Основи Kotlin. Характеристика Kotlin. Основні типи даних. Змінні. Модифікатори доступу. Функції. Класи. Цикли. Складні типи даних Kotlin. Data class. Мульти декларації. Sealed class. Setter and Getter. Collections. List. Collections. Set. Collections. Map. Операції з колекціями.

Тема 9. Збереження та обробка даних у мобільних додатках.

Графічні елементи мобільних застосунків. Поняття про XML. Загальна структура інтерфейсу застосунку. Ресурсні файли. Використання графічних елементів на різних пристроях. Доступ та керування елементами екрана з коду. Опрацювання подій. ViewBinding. Зв'язок XML та Kotlin. Binding. Про події в цілому. Типи подій. Обробка подій з XML. Обробка подій у Kotlin файлі - клік кнопки. Activity та її життєвий цикл. Application. Manifest. Життєвий цикл Activity. Створення Activity. Context. Перехід з одного Activity на інше. Manifest. Життєвий цикл Application. Створення власного класу Application. EventListener та його типи.

Fragment та його життєвий цикл. Android jetpack components: navigation. Fragment і його життєвий цикл. Android Jetpack Navigation. Додавання Navigation graph в Activity. Робота зі списками. ListView, RecyclerView, Adapters. Glide. Обмежений та нескінченний список. Адаптери. RecyclerView. Ідея ViewHolder. DiffUtil. Paging (Пагінація). Glide.

Тема 10. Архітектура мобільних додатків.

Основи архітектури додатку. MVVM, ViewModel. LiveData, coroutine. Архітектура додатку. Паттерн MVVM. Зв'язок компонентів архітектури у кодї. Вихід до інтернету. Retrofit, JSON. Зберігання даних. Preference. Основи баз даних. Room. Realm. Загальна схема роботи з інтернетом з додатку. Gson. Retrofit. GraphQL та Apollo. Варіанти зберігання даних в ОС Android. Збереження даних у локальній базі даних. Realm.

Тема 11. Захист інформації в мобільних системах.

Особливості захисту інформації в мобільних пристроях. Основні погрози для мобільних пристроїв. Аналіз ризиків. Практичні аспекти захисту інформації у системах мобільного зв'язку стандарту GSM. Платформи безпеки мобільних ОС. Розділення коду і даних. Права доступу. Захист на етапі публікації мобільного додатка, підпис додатків. Мобільне протівірусне ПЗ. Захист інформації у мережах мобільного зв'язку. Особливості забезпечення безпеки, механізми автентифікації, цілісності, конфіденційності та анонімності. Практичні аспекти захисту інформації в системах мобільного зв'язку з кодовим розподілом каналів стандарту IS-95. Контроль доступу. Особливості захисту від помилок. Перешкодостійке кодування. Захист інформації у перспективних системах мобільного зв'язку.

Перелік навчальних робіт з дисципліни «Крос-платформне програмування»

Види робіт	К-сть балів
Лабораторне заняття №1. Тема: «Платформа .Net. Середовище розробки Visual Studio».	3
Лабораторне заняття №2. Тема: «Основні поняття мови програмування C#».	3
Лабораторне заняття №3. Тема: «Типи даних у C#».	3
Лабораторне заняття №4. Тема: «Створення проекту Windows Forms».	3

Лабораторне заняття №5. Тема «Програмування розгалужень за допомогою елементів керування RadioButton і CheckBox».	3
Лабораторне заняття №6. Тема: «Цикл з параметром. Випадаючий список. Нумерований список. Множинний вибір».	3
Лабораторне заняття №7. Тема: «Створення програм для обробки одновимірних масивів з виведенням даних в елементи управління екранних форм в C#».	3
Лабораторне заняття №8. Тема: «Створення програм для обробки двовимірних масивів з виведенням даних у таблицю на екранній формі в C#».	3
Лабораторне заняття №9. Тема: «Головне меню».	3
Лабораторне заняття №10. Тема: «Рядкові дані».	3
Лабораторне заняття №11. Тема: «Основи ООП».	3
Лабораторне заняття №12. Тема: «Успадкування та поліморфізм».	3
Лабораторне заняття №13. Тема: «Динамічні структури даних».	3
Лабораторне заняття №14. Тема: «Елементи керування ListView і TreeView».	3
Лабораторне заняття №15. Тема: «Черга. Елемент керування ToolStrip».	3
Лабораторне заняття №16. Тема: «Збереження налаштувань програми в XML-файл».	3
Лабораторне заняття №17. Тема: «Таймер і рядок статусу».	3
Лабораторне заняття №18. Тема: «Хеш-таблиці. Бінарні файли».	3
Модульний контроль	16
Виконання індивідуального завдання (СР)	10
Разом: Аудиторна робота	80
Самостійна робота (СР)	20
Всього 6 семестр:	100
Лабораторне заняття №19. Тема: «Знайомство з інтерфейсом GDI +».	4
Лабораторне заняття №20. Тема: «Побудова графіків функцій».	4
Лабораторне заняття №21. Тема: «Анімація графіки».	4
Лабораторне заняття №22. Тема: «Багатовіконний додаток».	4
Лабораторне заняття №23. Тема: «Розробка додатків для мобільних пристроїв».	4
Лабораторне заняття №24. Тема: «Обробка простих типів даних у програмах на мові Kotlin»	4
Лабораторне заняття №25. Тема: «Обробка складних типів даних у програмах на мові Kotlin».	4
Лабораторне заняття №26. Тема: «Збереження та обробка даних у мобільних додатках»	4
Лабораторне заняття №27. Тема: «Створення додатків, які використовують дві Activity»	6

Лабораторне заняття №27. Тема: «Створення додатка, який використовує бази даних»	8
Лабораторне заняття №27. Тема: «Засоби захисту інформації для мобільних пристроїв»	2
Модульний контроль	16
Виконання індивідуального завдання (СР)	16
Разом: Аудиторна робота	80
Самостійна робота (СР)	20
Всього 7 семестр:	100

КОНТРОЛЬ ТА КРИТЕРІЇ ОЦІНЮВАННЯ ЗНАТЬ СТУДЕНТІВ

При вивченні дисципліни використовуються наступні форми контролю знань студентів: поточний; модульний; підсумковий.

Поточний контроль передбачає перевірку теоретичних питань, самостійної роботи, практичних робіт та усне опитування по кожній практичній роботі. По даному виду контролю оцінювання знань здійснюється у відповідності до бального розподілу наведеного в попередній таблиці.

Модульний контроль передбачає виконання модульної контрольної роботи. Всі завдання оцінюються в 16 балів. Перше завдання (теоретичне) – 4 бали, друге завдання (практичне) – 6 балів, третє завдання (практичне) – 6 балів.

Формою підсумкового контролю є екзамен. Екзаменаційна оцінка (100 балів) є результатом виконання двох теоретичних питань (2 x 20 балів = 40 балів) та практичного завдання (60 балів).

Результуюча оцінка з дисципліни визначається як середня від балів набраних протягом семестру та отриманих на іспиті.

СПИСОК РЕКОМЕНДОВАНИХ ДЖЕРЕЛ

Основний:

1. Griffiths I. Programming C# 8.0: Build Windows, Web, and Desktop Application. 1st Edition. O'Reilly, 2020. – 802 p.
2. Horton J. Android Programming with Kotlin for Beginners: Build Android apps starting from zero programming experience with the new Kotlin programming language. Packt Publishing Ltd, 2019.
3. Давидов М.В., Демчук А.Б., Лозинська О.В. Програмне забезпечення мобільних пристроїв: навчальний посібник. Львів: Видавництво «Новий Світ-2000» 2020. 218 с.
4. Коноваленко І.В., Марущак П.О. Платформа .NET та мова програмування C# 8.0 : навчальний посібник. Тернопіль : ФОП Паляниця В. А., 2020. 320 с.
5. Skeet J. C# in depth. Manning, 2019. 528 p.