



**КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ 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	Краскевич Валерій Євгенович
Науковий ступінь	Доктор технічних наук
Вчене звання	Професор
Посада	Професор 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/NjY4NQ==/bf27ad9293fa2bb6f9b2c3031d4b6e4a.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;
- Студент, який пропустив практичне заняття, самостійно вивчає матеріал (при виникненні питань може звертатися за консультацією згідно розкладу консультацій викладачів оприлюдненого на сайті кафедри) за наведеними джерелами, виконує завдання і дає його викладачу.
- 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	2022-2023,2023-2024
Факультет	Факультет інформаційних технологій
Курс	3,4
Семестр	6,7
Освітній ступінь	Бакалавр
Галузь знань	12 «Інформаційні технології»
Спеціальність	122 «Комп'ютерні науки»
Загальна характеристика	Кількість годин –360 Кількість кредитів – 12 Види занять: лекції, лабораторні, самостійна робота. Співвідношення аудиторних годин і годин самостійної роботи - 158/212 Мова викладання – українська Форма викладання – очна
Підсумковий контроль	Екзамен
Програмне забезпечення	Платформа .Net, Visual Studio
Обладнання	Проектор, комп'ютерна техніка із встановленим програмним забезпеченням та доступом до мережі Інтернет.
Необхідні попередні дисципліни	«Вступ до комп'ютерних наук», «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	Метою 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. Завданням 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	ЗК 2 Здатність застосовувати знання у практичних ситуаціях ЗК 3 Знання та розуміння предметної області та розуміння професійної діяльності
Фахові компетентності (результати навчання)	СК 8 Здатність проектувати та розробляти програмне забезпечення із застосуванням різних парадигм програмування: узагальненого, об'єктно-орієнтованого, функціонального, логічного, з відповідними моделями, методами й алгоритмами обчислень, структурами даних і механізмами управління СК 10 Здатність застосовувати методології, технології та інструментальні засоби для управління процесами життєвого циклу інформаційних і програмних систем, продуктів і сервісів інформаційних технологій відповідно до вимог замовника СК 12 Здатність забезпечити організацію обчислювальних процесів в інформаційних системах різного призначення з урахуванням архітектури, конфігурування, показників результативності функціонування операційних систем і системного програмного забезпечення СК 13 Здатність до розробки мережевого програмного забезпечення, що функціонує на основі різних топологій структурованих кабельних систем, використовує комп'ютерні системи і мережі передачі даних та аналізує якість роботи комп'ютерних мереж
Програмні результати навчання	ПР 9 Розробляти програмні моделі предметних середовищ, вибирати парадигму програмування з позицій зручності та якості застосування для реалізації методів та алгоритмів розв'язання задач в галузі комп'ютерних наук. ПР 13 Володіти мовами системного програмування та методами розробки програм, що взаємодіють з компонентами комп'ютерних систем, знати мережні технології, архітектури комп'ютерних мереж, мати практичні навички технології адміністрування комп'ютерних мереж та їх програмного забезпечення. ПР 14 Володіти мовами системного програмування та методами розробки програм, що взаємодіють з компонентами комп'ютерних систем, знати мережні технології, архітектури комп'ютерних мереж, мати практичні навички технології адміністрування комп'ютерних мереж та їх програмного забезпечення.

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, перерахування (enum), структури, масиви. Робота із рядками символів (тип string).

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

Створення консольного проекту. Розгалуження, оператори `?`, `if`, `switch`, тренарні 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 об'єктно-орієнтованого програмування у C#.

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`. Індикатори. 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.

Характеристики ОС (платформ) для мобільних пристроїв. Архітектура мобільних ОС (платформ), недоліки і переваги. Функції і можливості мобільних ОС (платформ). Огляд сучасних інтегрованих систем програмування для мобільних пристроїв. Інтегроване середовище системи програмування `Visual Studio`, `Eclipse`, `Qt Creator`, `Xcode` та ін. Екранні емулятори мобільних пристроїв. Етапи встановлення середовища розробки. Лексичні особливості використання мов програмування для мобільних пристроїв: типи даних, класифікація і представлення даних, класи та об'єкти, бібліотеки класів, управління пам'яттю, домовленості про імена, робота з колекціями. Технологія `Silverlight`. Структура та компоненти мобільного прикладення, файли ресурсів. Життєвий цикл мобільного додатка.

Тема 9. Збереження та обробка даних у мобільних додатках.

Робота з файлами та збереження користувальницьких налаштувань. Збереження стану додатків. Пошук даних і файлів. Збереження даних поза екземплярами. Передача даних між сторінками додатків та між додатками, сумісне використання даних. Управління налаштуваннями. Типи ресурсів, створення ресурсів, використання файлів ресурсів у кодї програми. Локалізація та глобалізація додатків.

Тема 10. Технології сенсорного вводу.

Види та характеристики сенсорних екранів. Типові задачі, що вирішує сенсорний ввід. Основи апарату сенсорного вводу. Обробка простих торкань, технологія multi-touch. Обробка жестів.

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.

Тема 12. Характеристики мобільних додатків.

Особливості додатків для мобільних пристроїв. Етапи розробки та впровадження мобільних програм. Визначення функціональних вимог до мобільного додатка. Розробка інтерфейсу користувача на основі сприйняття. Розробка моделі даних та комунікації. Вимоги до мобільного програмного коду. Тестування та розгортання мобільного додатка. Застосування мобільних пристроїв в автоматизації бізнес-процесів. Схеми використання мобільних пристроїв. Перспективи розвитку мобільних ОС (платформ).

Перелік навчальних робіт студентів та оцінки їх у балах з дисципліни «Крос-платформне програмування»

Види робіт	К-сть балів
Лабораторне заняття №1. Тема: «Платформа .Net. Середовище розробки Visual Studio»	6
Лабораторне заняття №2. Тема: «Основні поняття мови програмування C#»	6
Лабораторне заняття №3. Тема: «Типи даних у C#»	6
Лабораторне заняття №4. Тема: «Типи даних у C#»	6
Лабораторне заняття №5. Тема: «Створення проектів на C# та програмування алгоритмів, функцій»	6
Лабораторне заняття №6. Тема: «Створення проектів на C# та програмування алгоритмів, функцій»	6
Лабораторне заняття №7. Тема: Успадкування та поліморфізм»	6
Лабораторне заняття №8. Тема: «Основи ООП»	6
Лабораторне заняття №9. Тема: «Динамічні структури даних»	6

Лабораторне заняття №10. Тема: «Динамічні структури даних»	6
Лабораторне заняття №11. Тема: «Програмування з використанням компонент»	6
Лабораторне заняття №12. Тема: «Програмування з використанням компонент»	6
Лабораторне заняття №13. Тема: «Елементи керування»	6
Лабораторне заняття №14. Тема: «Елементи керування»	6
Лабораторне заняття №15. Тема: «Розробка додатків для мобільних пристроїв»	8
Лабораторне заняття №16. Тема: «Збереження та обробка даних у мобільних додатках»	8
Лабораторне заняття №17. Тема: «Збереження та обробка даних у мобільних додатках»	8
Лабораторне заняття №18. Тема: «Засоби сенсорного вводу»	8
Лабораторне заняття №19. Тема: «Засоби захисту інформації для мобільних пристроїв»	8
Лабораторне заняття №20. Тема: «Засоби захисту інформації для мобільних пристроїв»	8
Лабораторне заняття №21. Тема: «Характеристики мобільних додатків»	8
Модульний контроль	40
Виконання індивідуального завдання (СР)	60
Разом: Аудиторна робота	140
Самостійна робота (СР)	60
Всього:	200

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 20 балів. Перше завдання (теоретичне) – 4 бали, друге завдання (практичне) – 8 балів, третє завдання (практичне) – 8 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. Чакраборти А. Microsoft .NET Framework. Розробка професійних проектів / Чакраборти А. – К.: БХВ-Київ, 2003. – 896 с.
2. Голощапов А.А. Google Android: Системні компоненти і мережеві комунікації/ Голощапов А.А. – К.: БХВ-Київ, 2012. –384 с.
3. Грицюк Ю.І., Рак. Т.Є. Програмування мовою С++ / Грицюк Ю.І., Рак. Т.Є. – Л. : ЛДУ БЖД, 2011. – 292 с.